Modeling, Design and Optimization of a Lightweight Robotic Arm Toward Assisted Living Applications

Zhou, Lelai

Publication date:
2012

Document Version
Early version, also known as pre-print

Department of Mechanical and Manufacturing Engineering
Aalborg University, Denmark.
Special Report No. 78

Modeling, Design and Optimization of a Lightweight Robotic Arm Toward Assisted Living Applications

Ph.D. Thesis

by

Lelai Zhou

Department of Mechanical and Manufacturing Engineering, Aalborg University
Fibigerstraede 16, DK-9220 Aalborg East, Denmark
e-mail: lzh@m-tech.aau.dk

Typeset in LaTeX and printed in Aalborg, April 2012.

ISBN 87-91464-32-3

Preface

The thesis is submitted according to the requirements for the Doctor of Philosophy at the Faculty of Engineering and Science, Aalborg University. The work has been carried out during October 2008 to September 2011 at the Department of Mechanical and Manufacturing Engineering (M-Tech), AAU. The work is supervised by Associate Professor Shaoping Bai, AAU, and co-supervised by Professor Michael R. Hansen, University of Agder, Norway.

There are several people to whom I am very grateful for their supports. Firstly, I would like to thank Associate Professor Shaoping Bai for his splendid supervision and support. I also wish to thank my co-supervisor Professor Michael R. Hansen at University of Agder, Norway, for always sharing his expertise to my project.

I would like to thank Professor John Rasmussen for his support of the work in the field of biomechanics. In addition, I would like to thank Assistant Professor Michael Skipper Anderson for his help of modeling in AnyBody Modeling System.

The project is sponsored by Aalborg University and Det Obelske Familiefond (DOF). Funding from the Chinese Scholarship Council (CSC) is gratefully acknowledged.

Aalborg, November 2011

Lelai Zhou

Abstract

The prospect of utilizing lightweight robots in human's daily lives has great appeal. For example, lightweight robotic arms mounted on mobile platforms can be used to assist elderly, disabled or immobilized people either on a daily basis or as part of rehabilitation and training programs. The demand for lightweight robotic arms in this context is, in many ways, intuitive, since it is replacing the human arm. The human arm payload-to-weight ratio is clearly that of an extreme lightweight robot. In order to meet the lightweight criteria, design issues such as joint configurations, selection of motors and gearboxes, structural dimensions, and most importantly, the robot mass, have to be considered. Utilizing systematic optimization methods to reduce the weight, while other issues are also considered, is a suitable approach to handle such a complex task.

The objective of the project is to design a lightweight robotic arm for people assistance applications. As a result, methods for optimal design of lightweight robotic arms have been developed and implemented for a 5 degrees of freedom topology. Both virtual and real prototyping have been carried out, and modeling has covered such disciplines as multibody kinematics and dynamics, drive-train dynamics and structural finite element. To make the robot lightweight, the optimization method is extensively utilized to minimize the mass of the robotic arm.

The optimization of the robotic arm is conducted at three different levels, with the main objective to minimize the robot mass. At the first level, only the drive-train of the robotic arm is optimized. The design process of a robotic arm is decomposed into selection of components for the drive-train to reduce the weight. The drive-train in each joint is modeled so that the calculated torque of each joint can be easily referenced to the rotor of motor. Three constraints are formulated on the performance of motors, and three constraints are on gearboxes. A co-simulation platform is developed, that returns the optimal selection of motors and gearboxes for each joint.

At the second level, kinematic data is combined with the drive-train in the optimization. For this purpose, a dynamic model of the robot

is required. Constraints are formulated on the motors, gearboxes and kinematic performance. The integrated optimization can reduce the weight further than the method with drive-train optimization only. Case studies are conducted for different robot trajectories and different kinematic performance constraints.

At the third level, a systematic optimization approach is developed, which contains design variables of structural dimensions, geometric dimensions and drive-train components. Constraints are formulated on the stiffness and deformation. The stiffness and deformation of the arm are calculated through FEA simulation. A trajectory generator has been developed to evaluate different trajectories. A group of four trajectories is chosen to conduct kinematic and dynamic simulation of the robotic arm.

In all the three levels, the problems of optimization are solved by a non-gradient optimization algorithm, the Complex method. The optimizations in these levels reduce the weight of an initial reference design by 38%, 41%, and 50% respectively. A prototype of the robotic arm is designed and built in the project. The prototype weighs 14 kg with a payload capacity of 5 kg, and is controlled via software developed in the project dedicated to the prototype.

The control strategy used to control the motion of a human arm could perhaps be utilized in the control of the designed robotic arm. A preliminary study on the optimal control of lightweight robotic arms has been conducted. The main control strategy has been to employ energy-optimal trajectories so as to minimize the required energy consumption by comparing with the human arm. As part of this study, both a musculoskeletal model and an analytical simplified human arm model have been developed. Comparative studies have been conducted, while the two proposed models are similar to a reference model in predicting energy-optimal trajectories. The analytical model is the most efficient among the three models.

Publications

Parts of the work have been published in or submitted to peer-reviewed scientific journals and international conferences.

Refereed Journal Papers

1. Lelai Zhou, Shaoping Bai, and Michael R. Hansen, "Design Optimization on the Drive Train of a Light-Weight Robotic Arm", *Mechatronics*, vol. 21, no. 3, pp:560-569, 2011. doi:10.1016/j.mechatronics.2011.02.004

2. Lelai Zhou, Shaoping Bai, Michael R. Hansen, and John Rasmussen, "Modeling of Human Arm Energy Expenditure for Predicting Energy Optimal Trajectories", *Modeling, Identification and Control*, vol. 32, no. 3, pp:91-101, 2011. doi:10.4173/mic.2011.3.1

3. Lelai Zhou, Shaoping Bai, and Michael R. Hansen, "Integrated Dimensional and Drive-Train Design Optimization of a Light-Weight Anthropomorphic Arm", *Robotics and Autonomous Systems*, vol. 60, no. 1, pp:113-122, 2012. doi:10.1016/j.robot.2011.09.004

4. Lelai Zhou, Shaoping Bai, and Michael R. Hansen, "An Integrated Approach to the Design and Optimization of Lightweight Robotic Arms", *IEEE Transactions on Robotics*, (submitted in July, 2011)

Conference Papers

1. Lelai Zhou, Shaoping Bai, and Michael R. Hansen, "Design and Kinematics of a 5-DOF Light-Weight Anthropomorphic Robotic Arm", *Proceedings of the 22nd Nordic Seminar on Computational Mechanics*, Aalborg, Denmark, pp. 205-208, October 2009

2. Lelai Zhou, Michael R. Hansen, and Shaoping Bai, "Drive Train Design Optimization of a 5-DOF Light-Weight Robotic Arm", *The 1st Joint International Conference on Multibody System Dynamics*, Lappeenranta, Finland, pages 10, May 2010

3. Lelai Zhou, Shaoping Bai, and Michael R. Hansen, "Integrated Design Optimization of a 5-DOF Assistive Light-weight Anthropomorphic Arm", *IEEE 15th International Conference on Advanced Robotics*, Tallinn, Estonia, pp. 659-664, June 2011.

Contents

1

General Introduction

The thesis presents research work on the modeling, design and optimization of a lightweight robotic arm for people assistance applications. In the design process, novel optimization methods have been developed to reduce the mass of the whole robotic arm.

1.1 Background

Lightweight robots are robots especially designed for mobility and interaction with a priori unknown environments and with humans. These applications pose the requirements of a lightweight design with high payload-to-weight ratio and high level of safety. The lightweight anthropomorphic arm is a new kind of robotic manipulators for people assistance applications in human and robot coexistence environments. The robotic arm has the potential to assist elderly and disabled people in their daily living. Many of these people are not capable of dealing with their own living tasks, such that they need round-the-clock assistance, which is very demanding on the resources of the surroundings. For this reason, these people would benefit significantly from a safe anthropomorphic robotic arm mounted on an electric wheelchair or a table and performing simple tasks such as picking and placing a bottle of water or holding a book.

The anthropomorphic robotic arm has to be lightweight. Industrial manipulators, which are normally heavy, rigid, suitable for operations with high speed and precision, are dangerous to humans and cannot be used for people assistance applications. The anthropomorphic robotic arm, however, can coexist and interact with humans. To achieve this goal, the arm has to be lightweight, compact, and energy efficient. This can be accomplished with optimization methods which are able to take multiple design issues into considerations simultaneously.

1.2 Lightweight robots

An application of lightweight robot in people assistance applications is demonstrated in Fig. 1.1. Some robotic systems relevant to this kind of assistive systems have been reported, such as FRIEND-I and FRIEND-II [1], KARES II [2], RAPTOR [3].

Figure 1.1: A lightweight anthropomorphic robotic arm mounted on an electric wheelchair.

Different concepts have been utilized in order to achieve lightweight design (see Fig. 1.2). DLR's robotics lab designed a 7-dof torque-controlled lightweight robotic arm with a payload-to-weight ratio of 1:1, while its own weight is 14 kg [4]. It is interesting to notice that this arm follows the architecture of typical industrial manipulators. To overcome the pyramidal effect, the DLR's arm is built with customized motors, Harmonic Drive gears, electronics, and linkages of carbon fiber material. To ensure dynamic control of the DLR arm, torque sensors are mounted on the flex spline component of the Harmonic Drive gear and therefore measure the joint torques acting on the links. An additional bearing decouples the disturbing forces and torques.

Kinova technology released a novel 6-dof lightweight robotic arm named JACO [5]. The JACO arm weighs 5 kg with a payload capacity of 1.5 kg.

A 5-dof self-containing lightweight manipulator was presented in [6] with a portable docking concept from wheelchair to docking stations in the room. The robot is capable of fitting into a wide range of adapted environments, which has significant implications for the care of the disabled and elderly people with special needs. The modularity of the system makes it possible for the system to grow as the level of disability of the user changes. The MANUS manipulator weighs only 9 kg, which is designed to be mounted on an electric wheelchair [7]. The 6-dof MANUS arm can be controlled by a keypad, a joystick or even by single-button control.

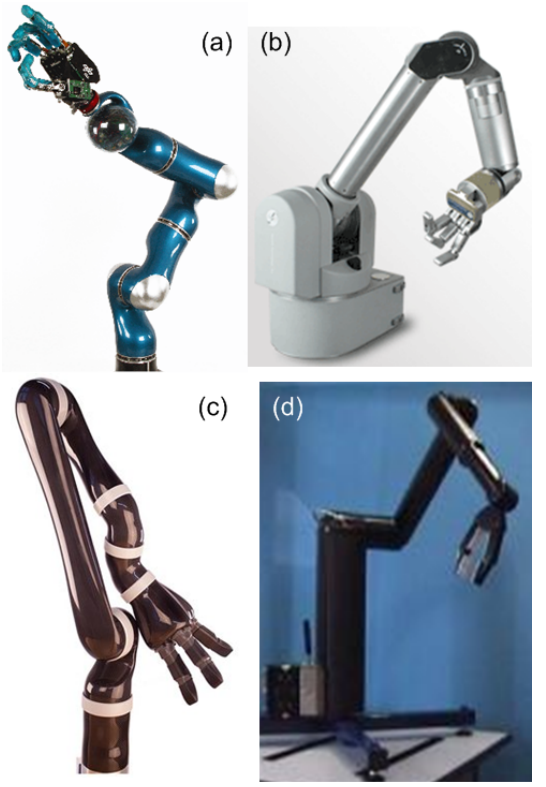


Figure 1.2: Selected lightweight robotic arms, (a) DLR arm, (b) WAM, (c) JACO, (d) MANUS.

The whole arm manipulator (WAM) developed originally at MIT and commercialized by Barrett Technology is a 7-dof lightweight arm driven by cable and cylinder transmissions [8]. The WAM arm has been designed with emphasis on transmissions, zero backlash, low friction and low inertia,

endowing the WAM with good open-loop backdriveability. Two key benefits of the high backdriveability are: (1) motion control through joint torque control, which enables the intrinsic sensing of forces over the whole arm and makes it inherently safe to humans; (2) operation directly in the Cartesian domain without the need for inverse kinematics calculations. Ananiev et al. [9] designed a 6-dof reconfigurable lightweight robotic arm mounted on a mobile robot. There are also some notable robotic arms assembled on humanoid robots including WABIAN-2 from Waseda University [10], Asimo of Honda [11], ARMAR from Karlsruhe University [12], Domo from MIT [13]. Out of them, Asimo and Domo have arms of 5 dof, while WABIAN-2 and ARMAR have arms of 7 dof.

1.3 Robot optimization

The design of lightweight robots faces many challenges, varying from power supply, actuators, power transmission and structural parts. New methods are required to meet these design challenges for lightweight designs. Optimization plays a key role in the design process.

1.3.1 Drive-train optimization

Different methods for optimal selection of motor and gear in mechatronic systems have been proposed. A method of finding the best motor/gear ratio combination for any given load with respect to weight, size, peak power, torque and efficiency was presented in [14]. Including the gearhead inertia and efficiency in the design process can give a large difference in the motor/gear ratio selection. Pasch and Seering proposed the optimal transmission ratio $n_m/n_l = \sqrt{J_l/J_m}$ and the optimal duration of motion between two points of velocity-limited systems [15]. A general method of motor and gearbox selection and optimization of servo drive system was introduced in [16, 17]. The method automated the solution procedure for the servo drive design problem by virtue of the normalization of torques, velocities, and transmission ratios. Moreover, the selection criteria separated the motor characteristics from the load characteristics and its graphical representation facilitated the feasibility check of a certain drive and the comparison between different systems. These methods above are applicable to the design of a single joint combining a motor and a gearbox, and they do not address the discrete nature of the selection process.

For design of robotic drive train consisting of multiple joints, the challenge

is that not only the characteristics of motor and gearbox at a single joint, but also the dynamics of the robot should be taken into account. An early attempt on drive-train design optimization can be found in [18], in which Chedmail and Gautier proposed a method for the optimum selection of robot actuators to minimize the total mass of all actuators. The modeling of the system took into account the inertia of the links and actuators, viscous and Coulomb friction effects, and the thermal model of the actuators as well.

A design optimization method on the drive-train of two joints was proposed for an industrial manipulator [19]. It presented an optimization strategy for finding the trade-offs between cost, lifetime, and performance when designing the drive-train for new robot concepts. The method was illustrated with an example in which the drive trains of two principal axes of a six-axis serial manipulator were designed. The method also included the trajectory generation of the robotic arm in the design loop to evaluate the impact of the trajectory.

A simulation environment called Modelica with robot optimization facilities was presented in [20], where the parameters of a controller and the drive-train were tuned by a multi-criteria parameter optimization method to improve the system dynamics. Dynamic loading criteria were addressed in the selection of actuators for optimal dynamic performance [21]. Dynamic performance was measured in terms of the robot's ability to accelerate the end-effector and to apply forces to the environment with given limitations on actuator torques. Dynamic capability equations [21] were used to model the relationship between actuator torque capacities and the acceleration and force capabilities. The selection of drive-train configuration for gear coupled manipulators was optimized for optimum dynamic performance in [22]. The kinematic structure of the gear coupled manipulator was described by an equivalent open-loop chain driven by mechanical power transmission lines. Gear ratios, location of actuators and details of mechanical transmission lines were determined according to the optimization.

1.3.2 Dimensional optimization

Dimensional optimization can contribute to the improvement of robotic performance, either kinematic performance or dynamic one. An integrated structure-control design optimization method of a two-link flexible robot arm was presented, where the structural and control parameters were optimized simultaneously [23]. The method used a genetic algorithm and the performance was compared with that of an arm with uniform

links and an optimized control system. The simultaneous optimization yielded a design with higher bandwidth and less weight of the arm system. An optimal design of manipulator parameter using an evolutionary optimization method was proposed in [24], in which a modification in differential evolution optimization technique was proposed to incorporate the effect of noises in the optimization process and obtain the optimal design of a manipulator. An optimum robot design method based on a specified task was proposed [25], in which dimensions were optimized based on dynamic analysis. Three evolutionary techniques were applied to minimize the torque required to perform the defined motion subject to constraints on link parameters and the end-effector deflection. A method for optimal dynamic performance was reported in [26], which consists of a parameter optimization using the motion time along the path as the cost function. Kinematic dimensions and actuator parameters were optimized through the dynamic optimization in manipulator design [27]. The dynamic model of the end-effector in operational space formulation was used to establish the relationship between joint torques and end-effector acceleration in the form of transformation matrices.

1.3.3 Structural optimization

Lightweight design of structural parts may lead to a significant reduction in the weight of the robot. Regarding structural optimization, finite element analysis (FEA) is widely used. FEA was utilized to conduct structural topology optimization in the design of humanoid robots [28]. Multibody system simulation (MBS) was employed to investigate the dynamics of the robot. By integrating MBS simulation into structural optimization processes, components in mechatronic systems could be optimized regarding the interaction between parts of mechanical properties and the overall system dynamics. FEA based design optimization was conducted on a 2-dof robot to minimize the vibration frequency [29]. The optimized design was compared with an experimental investigation of the structural vibration frequencies obtained on the actual manipulator. The utilization of FEA in robotic arm design and structural optimization can be found in [30].

The above robotic optimization technologies are summarized in Table 1.1.

Table 1.1: Overview of the robotic optimization technology.

No.	Objective	Design Variables	Constraints	Optimization Algorithm	Ref.
1	total mass	motor mass	minimize motor torques	KTNC[1]	[18]
2	mass, cost	motor torques	motor, joint dynamics	Complex	[19]
3	control performance	controller, drive-train	dynamics	Pattern search, GA[2]	[20]
4	dynamic performance	motor selection	motor torques	-	[21]
5	control performance	structural dimension	control	GA	[23]
6	kinematic and dynamic performance	geometric dimension, link mass	boundary limits	GA	[24]
7	joint torques	structural dimension	stiffness, deflection	GA	[25]

[1] Kuhn-Tucker Necessary Conditions

[2] Genetic Algorithm

1.4 Optimization methods

Optimization of robots becomes a discrete problem when design variables of motors and gearboxes are included. To this kind of problem, non-gradient methods are more robust in locating the global optima and are applicable in a broader set of problem domains.

There is a great number of non-gradient optimization methods. The Complex method was developed by Box [31]. The genetic algorithm [32] developed by Holland was robust in finding the global optimum. Simulated annealing was then developed by Kirkpatrick [33]. This method makes very few assumptions regarding the function to be optimized, and therefore, it is quite robust with respect to irregular surfaces. Apart from these methods, there are also other promising techniques to conduct engineering optimization, for instance Tabu search method [34], response surface approximations [35] and particle swarm method [36].

The selection of methods depends on the problem and how well the method

fits the particular problem. This thesis is focused on non-gradient methods, as they are better suited for the design problems. Another advantage of non-gradient methods is that they do not require any derivatives of the objective function in order to find the optimum. Therefore they can be used as black-box methods without any need for continuous variation of the objective function. In this thesis, the objectives are results of computer simulations, such that the derivatives of the objective function are not explicitly known. In the work of the thesis, the Complex method [31] is extensively utilized, because it is easy to implement and parameterize [37]. Comparison of optimization methods has not been within the scope of this work.

1.5 Scope of work

The aim of this project is to design a lightweight robotic arm for people assistance applications. The works involved in the thesis are summarized in Fig. 1.3.

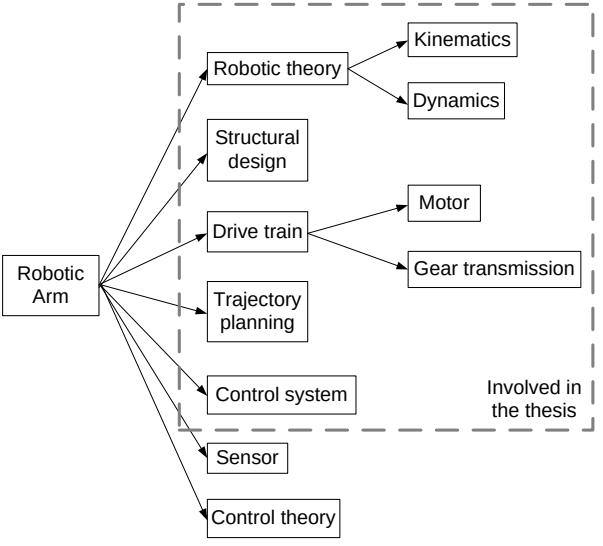


Figure 1.3: Different fields of technology involved in the architecture of robotic arms.

To reduce the weight of the robotic arm, optimization method will be developed in the design process. The approach of the project is summarized in the following steps:

1. Study basic kinematics and dynamics of robotic arms.
2. Model and design a 5-dof lightweight robotic arm.
3. Optimize the robotic arm to reduce the weight.
   (a) Optimize the drive-train components (motors and gearboxes).
   (b) Optimize the link lengths together with the drive-train.
   (c) Optimize the structural dimensions, link lengths and the drive-train.
4. Study control strategy for the anthropomorphic arm through simulation on human arm models.

Step 1: Basic knowledge of kinematics and dynamics of robotic arms is studied at this step.

Step 2: The robotic arm is designed with five joints (dof), following the inspiration from a human arm. The arm is configured with two joints at the shoulder, one at the elbow and two at the wrist. Forward kinematics and inverse kinematics are modeled for the robotic arm. Parameterized dynamic model of the robotic arm is built in Matlab™. The model will be used later as a simulation platform to conduct optimization.

Step 3(a): Drive-train design optimization is conducted on the robotic arm to reduce the total weight. The required driving torques of the robotic arm are determined from inverse kinematic and dynamic analysis within MSC.ADAMS™ [1]. The inverse kinematic and dynamic analysis of the robotic arm in ADAMS follows a so-called master-slave approach. The whole simulation system is developed as a simulation package, which will be called by the optimization program. In the dynamic simulation model, the mass of motors and gearboxes are parameterized, while the trajectory of the robotic arm is prescribed. For each updated motors and gearboxes, the required motor torques are calculated through the detailed drive-train model. The mass of distribution is updated during the optimization procedure. Modeling required at this step:

- Kinematics
- Dynamics

[1] MSC.ADAMS, a multibody dynamics simulation software.

- Drive-train modeling

Step 3(b): Geometric dimensions will be included together with drive-train in the robot optimization, aiming to reduce further the weight. The optimization of the geometric dimension can also lead to better kinematic performance through constraints on performance indices. Modeling required besides those in step 3(a) is *Global conditioning index*.

Step 3(c): The simulation and optimization platform developed in steps 3(a) and 3(b) is expanded to contain more design variables on the structural dimensions. To maintain the performance of the robot, stiffness and deformation are set as constraints. ANSYS™ Workbench is utilized to conduct finite element analysis on the whole body of the robotic arm. The outcomes of the optimization are the minimized weight, optimal structural dimensions, optimal geometric dimension and optimal drive-train components. Modeling required besides those in step 3(b) is *Finite element modeling of the whole robot*.

Step 4: Control strategy of the human arm could provide inspiration of controlling the designed robotic arm. The trajectory generation strategy of a human arm for planar motion is studied in this step. Two human arm models are built, one anatomical musculoskeletal model built in the AnyBody™ [2] Modeling System, and one analytical model. Metabolic energy costs are calculated with the two human arm models for parameterized arm trajectories. Energy optimal trajectories are obtained and compared to a reference model. The preliminary study of the human arm control strategy will lead to control method of the anthropomorphic arm in the future work.

1.6 Outline of thesis

The thesis is outlined as follows

Chapter 2 introduces the modeling and design of the robotic arm. Technical details for the prototype of the 5-dof robotic arm are described in this chapter. Kinematic and dynamic models of the robotic arm are built for conducting simulations. The modeling of kinematics and dynamics is implemented in Matlab, and presented with a numerical example.

Chapter 3 details design and optimization on the drive-train of a 5-dof lightweight robotic arm. Optimal selections of motors and gearboxes

[2] AnyBody, a musculoskeletal modeling and simulation software.

from a limited catalogue of commercially available components are done simultaneously for all joints of a robotic arm. Characteristics of the motor and gearbox, including gear ratio, gear inertia, motor inertia, and gear efficiency, are considered in the drive train modeling. A co-simulation method is developed for dynamic simulation of the arm. A design example is included to demonstrate the proposed design optimization method.

Chapter 4 describes an integrated method to optimize geometric dimension and drive-train of a robotic arm. The method addresses the influences of dimensions and characteristics of drive-trains in the design optimization. Constraints are formulated on the basis of kinematic performance and dynamic requirements, whereas the main objective is to minimize the total mass. Case studies are included to demonstrate the application of the optimization method in the design of assistive robots.

Chapter 5 details a new approach to the design and optimization of lightweight robotic arms, where robot kinematics, dynamics, drive-train design and strength analysis by means of FEA are considered. The approach makes use of a discrete parameterized robotic model, in which kinematic dimensions, structural dimensions, the motors and gearboxes from commercially available components are parameterized as design variables for design analysis and optimization. A co-simulation platform is developed, which couples Matlab with ANSYS Workbench simulations in one package. A group of four trajectories is used to conduct kinematics and dynamics simulation on the robotic arm.

Chapter 6 presents the study on the human arm motion to understand the strategy of generating energy optimal trajectories. The human arm moves in plane between groups of target points. One analytical model and one anatomical musculoskeletal model of the human arm are built. The two models also contain detailed model of the metabolic energy consumption. The arm trajectory is parameterized and formulated with the Fourier series, such that a group of trajectories can be evaluated with the models to estimate metabolic cost. The optimal trajectory is identified for each model and different target points. The performance of the two proposed models is compared to a reference model.

Chapter 7 concludes the thesis, with a summary of the findings of the work and contributions. Future works are also stated.

Note that Chapters 3 to 6 are written as full journal publications in this

dissertation. This causes some overlap between some chapters, but it also means that they can be read individually.

2

Design and Modeling of the Robotic Arm

The design and modeling of the robotic arm are the fundamental base of conducting optimization afterwards. This chapter introduces the mechanical design and technical details of the robotic arm, and also details the kinematic and dynamic modeling, followed by a numerical example.

2.1 Design of robotic arm

The design of the robotic arm is detailed with design considerations addressing the assisted living applications. Some technical details of the prototype are introduced as well.

2.1.1 Design considerations

The robotic arm is an anthropomorphic arm as it follows the nature design of a human arm. A human arm consists of seven dof, three at the shoulder, two at the elbow, and two at the wrist. The concept design of the robotic arm includes 5 dof, which reduces one dof in the shoulder and one in the elbow. When the concept design has been determined the physical properties from the design can be used to recalculate motions and torques. These can then again be used to redesign the first concept to a new and better one. This iteration process would be efficient to put inside an optimization procedure, where motors, gearboxes and structural design would be optimization factors.

The robotic arm will be used to handle daily tasks of people assistance applications. The total reach distance is 1 m (without the gripper), which is a bit longer than a human arm. The workspace of each joint is based on the corresponding joint workspace of the human arm. The range of each

joint workspace is listed in Table 2.1.

Table 2.1: Joint workspace of the robotic arm.

Joint i	Max Workspace	Constrained Workspace
1	$0 \sim 2\pi$	$0 \sim \pi$
2	$0 \sim 3\pi/2$	$0 \sim 3\pi/2$
3	$0 \sim 3\pi/2$	$0 \sim 3\pi/4$
4	$0 \sim 2\pi$	$0 \sim 2\pi$
5	$0 \sim 3\pi/2$	$0 \sim 3\pi/4$

The 5-dof robotic arm designed in this work adopts a modular approach. Harmonic Drive™ CPU series gearboxes are used as transmission elements and, simultaneously, as the mechanical joints for different dof. To increase the torque capabilities of Joints 1, 2 and 3, a second stage of gearhead is used between Harmonic Drive and the motor. Joints 2, 3 and 5 adopt the similar configurations. In Joint 4, only geared motor is used to transmit torque through bearing supported shaft, as shown in Fig. 2.1.

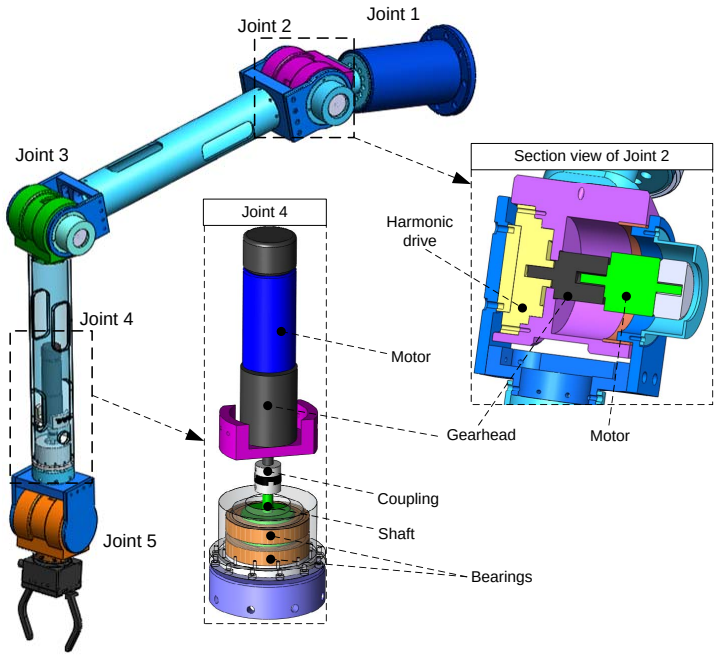


Figure 2.1: CAD rendering of a 5-dof lightweight anthropomorphic arm.

The structural parts are designed with the main criteria of minimizing the mass. Each joint is designed to contain the gearbox and motor in its housing in case of damage.

2.1.2 Technical details

The arm joints are driven by electrical motors, chosen among Maxon™ DC motors. The DC motors on the robot are from the Maxon RE series. The five motors have different power ratings, depending on the load in the joints; the used models are 60 W, 70 W, and 90 W. Table 2.2 shows the type of DC motor for each joint. For precise measurement of the angle during operation, an encoder is mounted on each motor. These are of the type Maxon motor Encoder MR, Type L, 1024 CPT, 3 Channels, and provide a relative measurement of the angle with 1000 pulses each turn.

Each motor is connected to an amplifier, namely, the Maxon EPOS2 24/5 amplifier, which has a built-in PID-controller, A/D converter, digital I/O, and CAN-bus. The built-in PID-controller enables speed and position control by use of the encoders mounted on the motors. Additionally, the EPOS amplifier also enables current control, i.e. torque control of the motor. The interface to the EPOS amplifier adopts the CANopen protocol which ensures that the measurements can be fetched and the control of the motors can be achieved digitally via the CANopen bus, i.e. the entire motor set-up is noise immune.

Table 2.2: Technical details of the robotic arm.

Joint i	Motor	Motor Power	Gearhead	Harmonic Drive	Total Ratio
1	RE 35	90 W	GP 42C	CPU 17A	430
2	EC-i 40	70 W	GP 32C	CPU 17A	1400
3	EC-i 40	70 W	GP 32C	CPU 14A	370
4	RE 30	60 W	GP 32A	-	51
5	EC-i 40	70 W	-	CPU 14A	100

2.2 Kinematics

The lightweight robotic arm is a serial manipulator consisting of several links connected in series by revolute joints. For a robot to perform a specific task, the location of the end-effector relative to the base should be established

first. This is called the position analysis problem. There are two types of position analysis problems: forward kinematics and inverse kinematics. In forward kinematics, the joint variables are given and the problem is to find the location of the end-effector. In inverse kinematics, the location of the end-effector is given and the problem is to find the joint variables necessary to bring the end-effector to the desired location.

2.2.1 Forward kinematics

Following Denavit-Hartenberg's convention [38], a Cartesian coordinate system is attached to each link of a manipulator, as shown in Fig. 2.2.

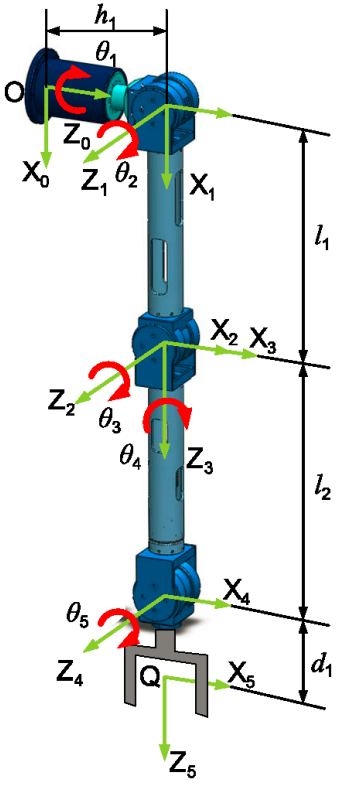


Figure 2.2: Robotic arm coordinate system.

Having established a coordinate system to each link of a manipulator, a 4×4 transformation matrix relating two successive coordinate systems can

be established.

$$ {}^{i-1}A_i = \begin{bmatrix} c\theta_i & -c\alpha_i s\theta_i & s\alpha_i s\theta_i & a_i c\theta_i \\ s\theta_i & c\alpha_i c\theta_i & -s\alpha_i c\theta_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.1} $$

According to the coordinate systems established in Fig. 2.2, the corresponding link parameters are listed in Table 2.3.

Table 2.3: D-H parameters of the robotic arm.

Joint i	α_i	a_i	d_i	θ_i
1	$\pi/2$	0	h_1	θ_1
2	0	l_1	0	θ_2
3	$\pi/2$	0	0	θ_3
4	$-\pi/2$	0	l_2	θ_4
5	$\pi/2$	0	d_1	θ_5

Substituting the D-H link parameters into Eq. (2.1), we obtain the D-H transformation matrices:

$$ {}^{0}A_1 = \begin{bmatrix} c\theta_1 & 0 & s\theta_1 & 0 \\ s\theta_1 & 0 & -c\theta_1 & 0 \\ 0 & 1 & 0 & h_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad {}^{1}A_2 = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & l_1 c\theta_2 \\ s\theta_2 & c\theta_2 & 0 & l_1 s\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.2} $$

$$ {}^{2}A_3 = \begin{bmatrix} c\theta_3 & 0 & s\theta_3 & 0 \\ s\theta_3 & 0 & -c\theta_3 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad {}^{3}A_4 = \begin{bmatrix} c\theta_4 & 0 & -s\theta_4 & 0 \\ s\theta_4 & 0 & c\theta_4 & 0 \\ 0 & -1 & 0 & l_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.3} $$

$$ {}^{4}A_5 = \begin{bmatrix} c\theta_5 & 0 & s\theta_5 & 0 \\ s\theta_5 & 0 & -c\theta_5 & 0 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.4} $$

The end-effector location is given by

$$ {}^0A_5 = \begin{bmatrix} u_x & v_x & w_x & q_x \\ u_y & v_y & w_y & q_y \\ u_z & v_z & w_z & q_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.5} $$

The loop-closure equation is obtained by

$$ {}^0A_5 = {}^0A_1 {}^1A_2 {}^2A_3 {}^3A_4 {}^4A_5 \tag{2.6} $$

Substituting Eqs. (2.2) through (2.4) into (2.6) yields the elements of ${}^0A_5$ as follows:

$$
\begin{aligned}
u_x &= c\theta_5(s\theta_1 s\theta_4 + c\theta_1 c\theta_{23} c\theta_4) - c\theta_1 s\theta_{23} s\theta_5 \\
u_y &= -c\theta_5(c\theta_1 s\theta_4 - s\theta_1 c\theta_{23} c\theta_4) - s\theta_1 s\theta_{23} s\theta_5 \\
u_z &= c\theta_{23} s\theta_5 + s\theta_{23} c\theta_4 c\theta_5 \\
v_x &= s\theta_1 c\theta_4 - c\theta_1 c\theta_{23} s\theta_4 \\
v_y &= -s\theta_1 c\theta_{23} s\theta_4 - c\theta_1 c\theta_4 \\
v_z &= -s\theta_{23} s\theta_4 \\
w_x &= s\theta_5(s\theta_1 s\theta_4 + c\theta_1 c\theta_{23} c\theta_4) + c\theta_1 s\theta_{23} c\theta_5 \\
w_y &= -s\theta_5(c\theta_1 s\theta_4 - s\theta_1 c\theta_{23} c\theta_4) + s\theta_1 s\theta_{23} c\theta_5 \\
w_z &= s\theta_{23} c\theta_4 s\theta_5 - c\theta_{23} c\theta_5 \\
q_x &= l_1 c\theta_1 c\theta_2 + l_2 c\theta_1 s\theta_{23} + d_1 \cdot (s\theta_1 c\theta_4 - c\theta_1 c\theta_{23} s\theta_4) \\
q_y &= l_1 s\theta_1 c\theta_2 + l_2 s\theta_1 s\theta_{23} - d_1 \cdot (c\theta_1 c\theta_4 + s\theta_1 c\theta_{23} s\theta_4) \\
q_z &= h_1 + l_1 s\theta_2 - l_2 c\theta_{23} - d_1 s\theta_{23} s\theta_4
\end{aligned}
$$

2.2.2 Inverse kinematics

Inverse kinematics of the robotic arm is also part of the position analysis. We present a more efficient method of solution by separating the wrist-center-position problem from the orientation problem. Note that the last two joint axes (Z_4 and Z_5) intersect at the wrist center point $\boldsymbol{C}$ as shown in Fig. 2.2. Hence rotations of the last two joints do not affect the position of $\boldsymbol{C}$. Figure 2.3 shows the end-effector coordinate system (X_5, Y_5, Z_5), the wrist center $\boldsymbol{C}$, and the vector relation between them.

The homogeneous position vector of the wrist center expressed in the end-effector coordinate system is

$$ {}^5\mathbf{p} = [0, 0, -d_1, 1]^T \tag{2.7} $$

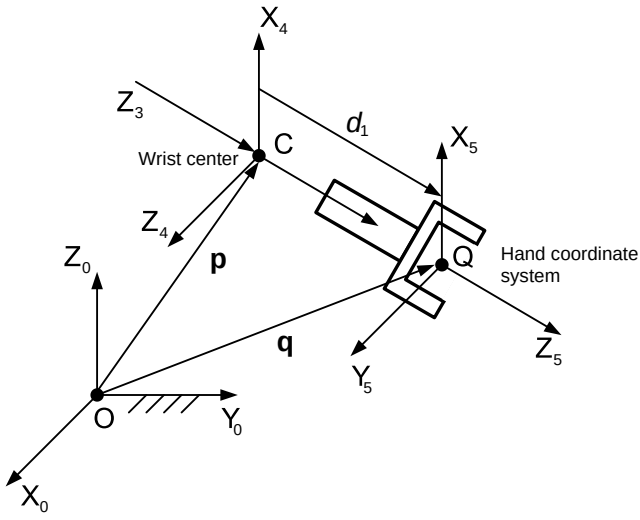


Figure 2.3: Hand coordinate system and wrist center position.

when expressed as in the base coordinate system, it becomes

$$ {}^0\mathbf{p} = {}^0A_5 {}^5\mathbf{p} = \begin{bmatrix} p_x \\ p_y \\ p_z \\ 1 \end{bmatrix} = \begin{bmatrix} q_x - d_1 w_x \\ q_y - d_1 w_y \\ q_z - d_1 w_z \\ 1 \end{bmatrix} \tag{2.8} $$

Hence, given the end-effector location, we can find the position of the wrist center point $\boldsymbol{C}$ with respect to the base coordinate system. Furthermore, we observe from Fig. 2.2 that the position of the wrist center $\boldsymbol{C}$ with respect to the link 3 coordinate system is given by

$$ {}^3\mathbf{p} = [0, 0, l_2, 1]^T \tag{2.9} $$

Transforming ${}^3\mathbf{p}$ into the base coordinate system, we obtain

$$ {}^0\mathbf{p} = {}^0A_3 {}^3\mathbf{p} \tag{2.10} $$

Equation (2.10) consists of three scalar equations in three unknowns. Hence the position and orientation of the inverse kinematics problem are decoupled. Solutions of the joint angles θ_1, θ_2, and θ_3 are presented, which are

$$\theta_1 = \arctan\left(\frac{p_y}{p_x}\right),\ 0 \le \theta_1 \le \pi \tag{2.11}$$

$$\theta_3 = \arctan\left(s\theta_3, c\theta_3\right),\ 0 \le \theta_3 \le \pi \tag{2.12}$$

where

$$p_x = q_x - d_1 w_x,\ p_y = q_y - d_1 w_y,\ p_z = q_z - d_1 w_z$$

$$\kappa_1 = l_1^2 + l_2^2,\ \kappa_2 = 2l_1 l_2,\ \kappa_3 = p_x^2 + p_y^2 + (p_z - h_1)^2,\ \kappa_2^2 - (\kappa_3 - \kappa_1)^2 > 0$$

$$s\theta_3 = (\kappa_3 - \kappa_1)/\kappa_2,\ c\theta_3 = \sqrt{1 - \left((\kappa_3 - \kappa_1)/\kappa_2\right)^2}$$

Upon solved θ_1 and θ_3, θ_2 is obtained as

$$\theta_2 = \arctan\frac{\mu_1\eta_2 + \eta_1\zeta_1}{\mu_1\eta_1 - \zeta_1\eta_2} \tag{2.13}$$

with

$$\mu_1 = l_1 + l_2 s\theta_3,\ \zeta_1 = l_2 c\theta_3,\ \eta_1 = p_x c\theta_1 + p_y s\theta_1,\ \eta_2 = p_z - h_1$$

Once θ_1, θ_2, and θ_3 are solved, ${}^0A_3$ is completely known. The remaining joint angles can be found by multiplying both sides of Eq. (2.6) by $({}^0A_3)^{-1}$

$${}^3A_5 = ({}^0A_3)^{-1}{}^0A_5 \tag{2.14}$$

${}^0A_3$ can be calculated by

$${}^0A_3 = {}^0A_1{}^1A_2{}^2A_3 = \begin{bmatrix} c\theta_1 c\theta_{23} & s\theta_1 & c\theta_1 s\theta_{23} & l_1 c\theta_1 c\theta_2 \\ s\theta_1 c\theta_{23} & -c\theta_1 & s\theta_1 s\theta_{23} & l_1 s\theta_1 c\theta_2 \\ s\theta_{23} & 0 & -c\theta_{23} & h_1 + l_1 s\theta_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.15}$$

While ${}^3A_5$ could be obtained as

$${}^3A_5 = {}^3A_4{}^4A_5 = \begin{bmatrix} c\theta_4 c\theta_5 & -s\theta_4 & c\theta_4 s\theta_5 & -ds\theta_4 \\ s\theta_4 c\theta_5 & c\theta_4 & s\theta_4 s\theta_5 & dc\theta_4 \\ -s\theta_5 & 0 & c\theta_5 & l_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.16}$$

We note that the elements on the right-hand side of Eq. (2.14) are known, i.e.,

$$({}^0A_3)^{-1}{}^0A_5 =$$

$$\begin{bmatrix} u_x c\theta_1 c\theta_{23} + u_y s\theta_1 c\theta_{23} + u_z s\theta_{23} & v_x c\theta_1 c\theta_{23} + v_y s\theta_1 c\theta_{23} + v_z s\theta_{23} \\ u_x s\theta_1 - u_y c\theta_1 & v_x s\theta_1 - v_y c\theta_1 \\ u_x c\theta_1 s\theta_{23} + u_y s\theta_1 s\theta 23 - u_z c\theta_{23} & v_x c\theta_1 s\theta_{23} + v_y s\theta_1 s\theta_{23} - v_z c\theta_{23} \\ 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} w_x c\theta_1 c\theta_{23} + w_y s\theta_1 c\theta_{23} + w_z s\theta_{23} & q_x c\theta_1 c\theta_{23} + q_y s\theta_1 c\theta_{23} + (q_z - h_1) s\theta_{23} \\ w_x s\theta_1 - w_y c\theta_1 & q_x s\theta_1 - q_y c\theta_1 \\ w_x c\theta_1 s\theta_{23} + w_y s\theta_1 s\theta_{23} - w_z c\theta_{23} & q_x c\theta_1 s\theta_{23} + q_y s\theta_1 s\theta_{23} - (q_z - h_1) c\theta_{23} \\ 0 & 1 \end{bmatrix} \tag{2.17}$$

only the rotation part of Eq. (2.14) is needed for computation of the last two joint angles. The rotation matrices are given by the upper 3×3 sub-matrices of Eqs. (2.15) and (2.16), respectively.

Equating the $(3,3)$ elements of Eqs. (2.16) and (2.17) yields

$$\theta_5 = \arccos(r_{33}),\ 0 < \theta_5 < \pi \tag{2.18}$$

where $r_{33} = w_x c\theta_1 s\theta_{23} + w_y s\theta_1 s\theta_{23} - w_z c\theta_{23}$. Corresponding to each solution set of θ_1, θ_2, and θ_3, Eq. (2.18) yields (1) two real roots if $|r_{33}| < 1$, and (2) $\theta_5 = 0$ or π if $|r_{33}| = 1$. When $\theta_5 = 0$ or π, the end-effector coordinate system Z-axis, Z_5, is in line with the fourth joint axis, Z_3, and the wrist is said to be in a singular configuration. The condition $|r_{33}| > 1$ cannot physically arise.

Assuming that $s\theta_5 \neq 0$, we can solve θ_4 as follows. Equating the $(1,3)$ element of Eqs. (2.16) and (2.17) yields

$$c\theta_4 = \frac{w_x c\theta_1 c\theta_{23} + w_y s\theta_1 c\theta_{23} + w_z s\theta_{23}}{s\theta_5} \tag{2.19}$$

Equating the $(2,3)$ element of Eqs. (2.16) and (2.17) yields

$$s\theta_4 = \frac{w_x s\theta_1 - w_y c\theta_1}{s\theta_5} \tag{2.20}$$

Hence, corresponding to each solution set of θ_1, θ_2, θ_3, and θ_5, Eqs. (2.19) and (2.20) yield a unique solution of θ_4,

$$\theta_4 = \arctan(s\theta_4, c\theta_4) \tag{2.21}$$

2.3 Dynamics

Dynamics deals with the forces and torques that cause the motion of a system of bodies. Analogously to direct and inverse kinematics analysis, there is direct and inverse dynamic analysis.

2.3.1 Jacobian matrix

The joint angular velocity can be calculated with the Jacobian matrix

$$\dot{\boldsymbol{\theta}} = \mathbf{J}^{-1}\mathbf{v}_{ef} \tag{2.22}$$

where $\dot{\boldsymbol{\theta}} = [\dot{\theta}_1, \dot{\theta}_2, \ldots, \dot{\theta}_n]^T$ denotes an n-dimensional (n denotes the number of dof) vector of the joint angular velocities, $\mathbf{J}$ is the Jacobian of the robotic arm, and $\mathbf{v}_{ef}$ the velocity of the end-effector.

For a revolute joint, the Jacobian matrix can be calculated by [39]

$$\mathbf{J} = [\mathbf{j}_1, \mathbf{j}_2, \ldots, \mathbf{j}_n]\,,\ \mathbf{j}_i = \begin{bmatrix} \mathbf{z}_{i-1} \times \mathbf{p}_{i-1} \\ \mathbf{z}_{i-1} \end{bmatrix} \tag{2.23}$$

where $\mathbf{z}_{i-1}$ and $\mathbf{p}_{i-1}$ are given by

$$\mathbf{z}_{i-1} = \mathbf{R}_{i-1}\,[0 \;\; 0 \;\; 1]^{\mathrm{T}}\,,\ \mathbf{p}_{i-1} = \mathbf{R}_{i-1}\mathbf{q}_{i-1} + \mathbf{p}_i \tag{2.24}$$

where $\mathbf{q}_{i-1} = [a_i \cos\theta_i, a_i \sin\theta_i, d_i]^{\mathrm{T}}$, $\mathbf{R}_{i-1}$ denotes the rotation matrix from the reference coordinate system to the $(i-1)$th coordinate system. The local coordinates of the end-effector are defined as $\mathbf{p}_n = [0, 0, 0]^{\mathrm{T}}$. When the desired end-effector velocity $\mathbf{v}_{ef}$ is given, the joint angular velocity can be solved by Eq. (2.22).

2.3.2 Inverse dynamics

The computation of the inverse dynamics is a prerequisite for evaluating any given design with given load and prescribed trajectory. Here we briefly recall the Lagrange-Euler formulation. The Lagrange equation is

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_i}\right) - \frac{\partial L}{\partial \theta_i} = \tau_i; \qquad i = 1, \ldots, n \tag{2.25}$$

where the Lagrangian $L = K - U = \sum_{i=1}^{n}(K_i - U_i)$. For the ith link, the kinetic energy K_i and the potential energy U_i are given by

$$K_i = \frac{1}{2}m_i\mathbf{v}_{c,i}^T\mathbf{v}_{c,i} + \frac{1}{2}\boldsymbol{\omega}_i^T\mathbf{I}_i\boldsymbol{\omega}_i; \quad U_i = m_i\mathbf{g}^T\mathbf{p}_{c,i} \tag{2.26}$$

where $\mathbf{v}_{c,i}$ denotes the linear velocity of the center of mass for link i, $\boldsymbol{\omega}_i$ is the angular velocity of the same link, and $\mathbf{I}_i$ is the inertia matrix of link i with respect to its center of mass. Moreover, $\mathbf{p}_{c,i}$ is the position vector of the center of mass for link i, measured in the reference coordinate system.

The governing equation of the arm motion can be written as

$$\mathbf{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \mathbf{V}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \mathbf{G}(\boldsymbol{\theta}) = \boldsymbol{\tau} \tag{2.27}$$

where $\mathbf{M}$ is the mass matrix, $\mathbf{V}$ is the vector of Coriolis and centrifugal terms of the links, $\mathbf{G}$ is the vector of gravitational forces, $\boldsymbol{\tau}$ is the vector of joint torques, and $\boldsymbol{\theta}$ is the vector of joint angles.

The mass matrix $\mathbf{M}$ can be calculated as

$$\mathbf{M} = \sum_{i=1}^{n} (\mathbf{J}_{v,i}^{\mathrm{T}} m_i \mathbf{J}_{v,i} + \mathbf{J}_{\omega,i}^{\mathrm{T}} \mathbf{I}_i \mathbf{J}_{\omega,i}) \tag{2.28}$$

where $\mathbf{J}_{v,i}$ and $\mathbf{J}_{\omega,i}$ are $3 \times n$ matrices. For revolute joint, the jth column vectors of $\mathbf{J}_{v,i}$ and $\mathbf{J}_{\omega,i}$ can be obtained by [39]

$$\mathbf{j}_{v,i}^{j} = \mathbf{z}_{j-1} \times \mathbf{p}_{c,i}^{j-1},\ \mathbf{j}_{\omega,i}^{j} = \mathbf{z}_{j-1}, \quad \text{for} \quad j \leq i \tag{2.29a}$$

$$\mathbf{j}_{v,i}^{j} = \mathbf{j}_{\omega,i}^{j} = [0 \quad 0 \quad 0]^T, \quad \text{for} \quad i < j \leq n \tag{2.29b}$$

where $\mathbf{p}_{c,i}^{j-1}$ is a position vector defined from the origin of the $j-1$ link frame to the center of mass of link i and expressed in the base frame. Moreover, m_i is the mass of the link i. For each link, its mass is found as

$$m_i = m_{s,i} + m_{m,i} + m_{g,i} \tag{2.30}$$

where $m_{s,i}$ is the mass of the arm structure, which is proportional to the link length. $m_{m,i}$ and $m_{g,i}$ are the masses of motor and gearbox for the ith joint. Both m_i and $\mathbf{I}_i$ vary with the selections of motors and gearboxes, and the link lengths as well.

2.3.3 A numerical example

The modeling of kinematics and dynamics is implemented in Matlab. One example simulation is executed on the 5-dof lightweight robotic arm to evaluate the modeling of kinematics and dynamics. The trajectory of the end-effector in the base coordinate system is defined as $X_{ef}(t) = 5+400(1-\cos(t))$, $Y_{ef}(t) = -990+800(1-\cos(t/2))$, and $Z_{ef}(t) = 280+250(\cos(t/2)-$

1), all with unit of mm. The Euler angles for the end-effector are given as $[0, \cos(t/180), 0]$, which implies the end-effector remains horizontal during the prescribed motion. The trajectory of the robotic arm is depicted in Fig. 2.4.

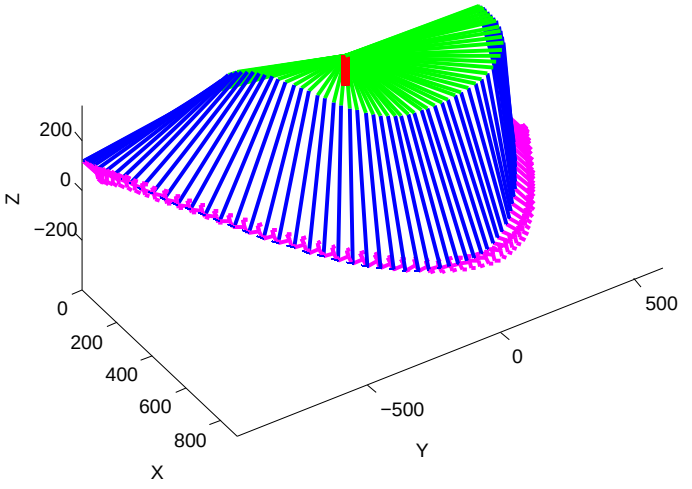


Figure 2.4: Trajectory of the robotic arm.

The joint angular velocities and accelerations are solved through Eq. (2.22). The solved results are depicted in Fig. 2.5 and Fig. 2.6.

Through the modeling of inverse dynamics, the solved joint torques are illustrated in Fig. 2.7.

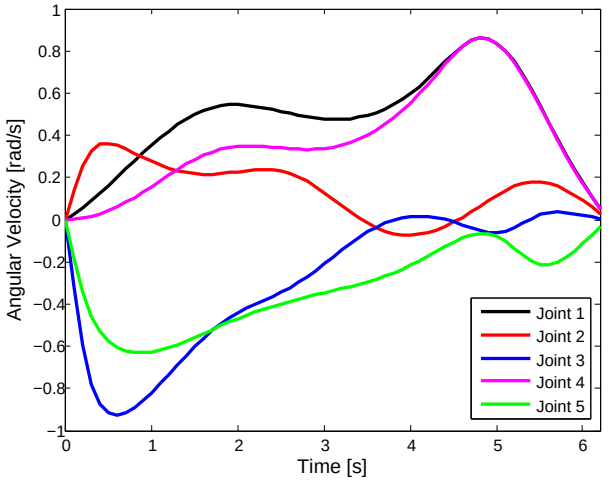


Figure 2.5: Angular velocities of the robotic arm.

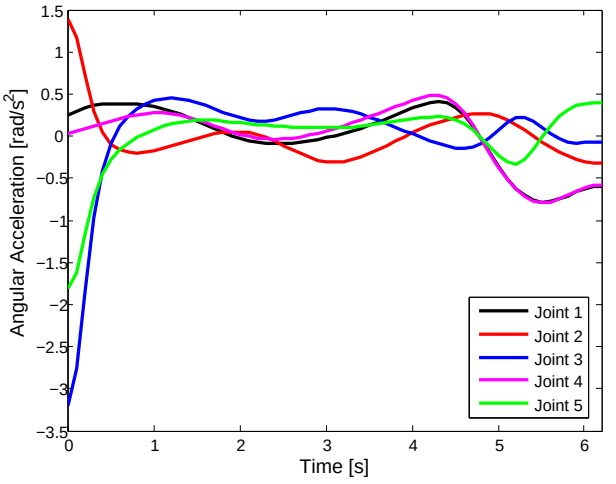


Figure 2.6: Angular accelerations of the robotic arm.

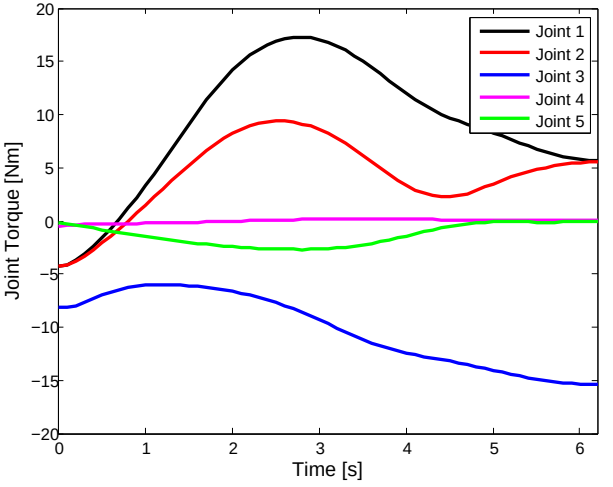


Figure 2.7: Joint torques of the robotic arm.

3

Article I

Design Optimization on the Drive Train of a Light-Weight Robotic Arm

The paper has been published as:

Lelai Zhou, Shaoping Bai, and Michael R. Hansen, "Design Optimization on the Drive Train of a Light-Weight Robotic Arm", *Mechatronics*, vol. 21, no. 3, pp:560-569, 2011. doi:10.1016/j.mechatronics.2011.02.004

Abstract

A drive train optimization method for design of light-weight robots is proposed. Optimal selections of motors and gearboxes from a limited catalogue of commercially available components are done simultaneously for all joints of a robotic arm. Characteristics of the motor and gearbox, including gear ratio, gear inertia, motor inertia, and gear efficiency, are considered in the drive train modeling. A co-simulation method is developed for dynamic simulation of the arm. A design example is included to demonstrate the proposed design optimization method.

3.1 Introduction

The drive train is the core part of a robot system, with significant impact on the cost and performance of the whole system. To achieve a light-weight design, drive train optimization plays a key role. A number of methods for motor and gear selection in mechatronic systems have been proposed. Pasch and Seering [15] studied maximizing the system acceleration by optimal selection of transmission ratio. Van De Straete et al. [16, 17] proposed a general method of motor and gearbox selection for optimization of servo drive system. The method automates the solution procedure for the servo drive design problem by virtue of the normalization of torques, velocities, and transmission ratios. Cetinkunt [40] proposed an optimization approach of balancing the high speed and precision in servo systems. Cusimano [41, 42] presented a procedure for optimal selection of an electrical motor and transmission. Roos et al. [14] proposed a method of finding the best motor/gear ratio combination for any given load with respect to weight, size, peak power, torque and efficiency. The methods above are applicable to the design of a single joint combining a motor and a gearbox, and they do not address the discrete nature of the selection process.

For design of robotic drive train consisting of multiple joints, the challenge is that not only the characteristics of motor and gearbox at a single joint, but also the dynamics of the robot should be taken into account, the latter varying with the selection of components and link dimensions. Furthermore, the optimization procedure adopted has to be capable of handling discrete design variables because the transmission is typically composed of commercially available components. Very few methods are available for the optimization of the entire drive train of a robot under constraint of available components. A method for the optimum selection of robot

actuators was proposed in [18], with objective to minimize the total mass of all the actuators under torque and temperature constraints. Pettersson and Ölvander [19] reported recently a method of design optimization, in which drive train for two joints were optimized for an industrial manipulator. The method is not applicable to selection of components from a catalogue. An evolutionary approach of optimization on robot configurations was reported in [43]. A simulation environment called Modelica with robot optimization characteristic was presented in [20], where the parameters of a controller can be tuned by a multi-criteria parameter optimization method to improve the system dynamics. DLR's 7-dof (degrees of freedom) torque-controlled light-weight robotic arm was built with customized motors and gearboxes to achieve a low weight [4]. Methods of robot optimization can also be found in [22, 44, 45], among others.

In this paper, an optimization method for drive train design of a light-weight robotic arm is proposed. The method is applicable to serial robotic arms, aiming at minimizing the arm weight. In the method, the optimization is carried out with a prescribed trajectory of the end-effector, generated within the robotic arm's workspace. Moreover, the inverse kinematic analysis was conducted in ADAMS to verify the trajectory is within the joint space. A dynamic model of the robotic arm is developed, upon which an optimization problem is formulated. A non-gradient optimization method, namely, the Complex [31], is implemented to run the optimization. The method is implemented on a co-simulation platform, where robotic dynamics is determined using MSC.ADAMS™, and the complex optimization is performed in Matlab™.

3.2 Conceptual design of a robotic arm

The light-weight robotic arm considered in this paper has five degrees of freedom (dof), with two dof at the shoulder, one at the elbow, and two at the wrist, as depicted in Fig. 3.1. The arm is designed for assisting elderly and handicapped people in daily living [46]. Light-weight design of such robotic arms is required for safety and energy efficiency.

In this design, harmonic drives are used as gearing elements. The motors and harmonic drive gearboxes are mounted inside the joints, while the axes of rotation coincide with the joint axes. The physical realization of Joint 2 is illustrated in Fig. 3.2. The same conceptual design is used for all 5 joints.

While the topology of the individual transmission is fixed, the motors

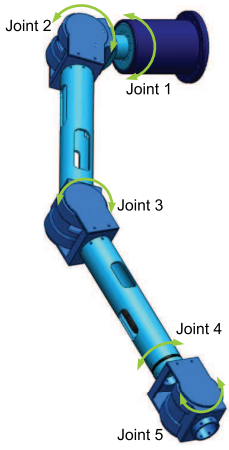


Figure 3.1: A 5-dof light-weight robotic arm.

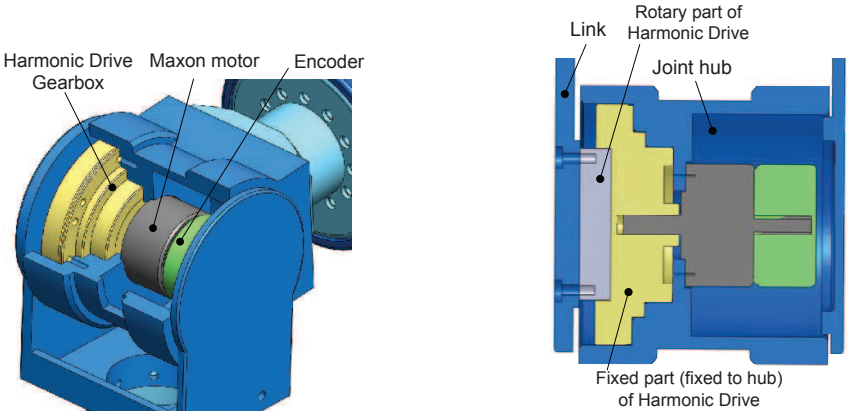


Figure 3.2: Joint mechanism for Joint 2, (a) 3D view, (b) section view.

can be chosen from either permanent magnet DC motors or brushless DC motors. Maxon™ motors are used in this study. The gearboxes are limited to Harmonic Drive™ backlash-free coaxial gears. Both components are considered appropriate for implementing the proposed design optimization method, that may easily accommodate a wider variety of gearboxes and motors. The arm structures are made of aluminium.

3.3 Kinematics and dynamics

3.3.1 Kinematics

The forward kinematics of the robotic arm is formulated based on the Denavit-Hartenberg convention [38]. A Cartesian coordinate system is attached to each link of the robotic arm, as shown in Figure 3.3. D-H parameters are defined as listed in Table 3.1. The detailed solution of the forward kinematics can be found in Appendix 1.

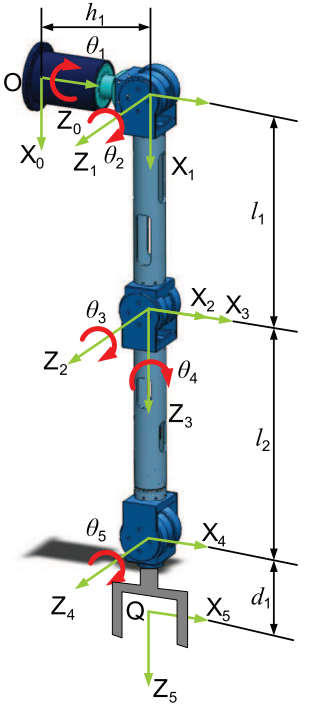


Figure 3.3: Robotic arm coordinate system.

For given locations of the end-effector, the joint variables are found by inverse kinematics. The method presented in [39] was adopted for this purpose. The detailed solution of the inverse kinematics can be found in Appendix 2.

3.3.2 Inverse dynamics

The computation of the inverse dynamics is a prerequisite for evaluating any given design with given load and prescribed trajectory. Here we briefly

Table 3.1: D-H Parameters of the Robotic Arm.

Joint i	α_i	a_i	d_i	θ_i
1	$\pi/2$	0	h_1	θ_1
2	0	l_1	0	θ_2
3	$\pi/2$	0	0	θ_3
4	$-\pi/2$	0	l_2	θ_4
5	$\pi/2$	0	d_1	θ_5

recall the Lagrange-Euler formulation, which is

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_i}\right) - \frac{\partial L}{\partial \theta_i} = \tau_i; \quad i = 1, \ldots, 5 \tag{3.1}$$

where the Lagrangian $L = K - U = \sum_{i=1}^{5}(K_i - U_i)$. For the ith link, the kinetic energy K_i and the potential energy U_i are given by

$$K_i = \frac{1}{2} m_i \mathbf{v}_{ci}^T \mathbf{v}_{ci} + \frac{1}{2} \boldsymbol{\omega}_i^T \mathbf{I}_i \boldsymbol{\omega}_i; \quad U_i = m_i \mathbf{g}^T \mathbf{p}_{ci} \tag{3.2}$$

where $\mathbf{v}_{ci}$ denotes the linear velocity of the center of mass for link i, $\boldsymbol{\omega}_i$ is the angular velocity of the same link, and $\mathbf{I}_i$ is the inertia matrix of link i with respect to its center of mass. Moreover, $\mathbf{p}_{ci}$ is the position vector of the center of mass for link i, measured in the reference coordinate system.

Substituting Eq. (3.2) into (3.1) produces equations of motion as

$$\mathbf{M}(\theta_i)\ddot{\theta}_i + \mathbf{v}(\theta_i, \dot{\theta}_i) + \boldsymbol{\xi}(\theta_i) = \boldsymbol{\tau} \tag{3.3}$$

where $\mathbf{M}$ is the mass matrix, $\mathbf{v}$ is the vector of Coriolis and centrifugal terms of the links, $\boldsymbol{\xi}$ is the vector of gravitational forces, and $\boldsymbol{\tau}$ is the vector of joint torques.

Equation (3.3) can be solved with different approaches [47, 48]. In this work, the dynamics solutions are found through ADAMS, which directly takes advantage of the accurate geometry and mass property of a CAD embodiment for computations. In the meantime, a Matlab solver adopted the recursive approach [49] was also developed for the purpose of comparison, which is discussed in Section 3.6.

3.3.3 Drive train modeling

Equation (3.3) yields the required joint torque $\tau(t)$, if the motion is prescribed. The motor torque for each joint can further be determined, as seen in Fig. 3.4. For the harmonic drive gearbox, the gear efficiency varies depending on the output torque. With the inertia of motor and gear, the required motor torque for the ith joint is derived as

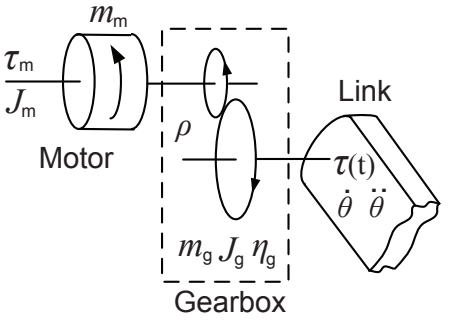


Figure 3.4: Schematic view of drive train model for a single joint.

$$\tau_{m,i} = \left\{ (J_m + J_g)\ddot{\theta}(t)\rho + \frac{\tau(t)}{\rho\eta_g} \right\}_i ; \quad i = 1, \ldots, 5 \tag{3.4}$$

where ρ_i is the gear ratio, $J_{g,i}$ is the gear inertia with respect to the input motor axis, $J_{m,i}$ is the motor inertia, and $\eta_{g,i}$ is the gear efficiency .

3.4 Formulation of design problems

The criteria for selecting motor and gearbox are applicable to each single joint, thus subscript *'i'* is omitted in this section for clarity.

3.4.1 Motor selection criteria

Motors for robotic arms are usually selected from two motor groups, brushed and brushless DC motors. In selecting motors, the following three constraints must not be violated:

Nominal torque limit. The nominal torque is the so-called maximum continuous torque. The root mean square (RMS) value τ_{rms} of the required motor torque τ_m has to be smaller than or equal to the nominal torque of the motor T_m

$$\tau_{rms} \leq T_m \tag{3.5}$$

where $\tau_{rms} = \sqrt{\frac{1}{\Delta t}\int_0^{\Delta t} \tau_m^2 dt}$, with Δt being the duration of a characteristic working cycle.

Stall torque limit. The stall torque is the peak torque of the motor. The required peak torque τ_p has to be smaller than or equal to the stall torque T_m^{max} of the motor

$$\tau_p \leq T_m^{max} \tag{3.6}$$

where $\tau_p = \max\{|\tau_m|\}$.

Maximum permissible speed limit. The maximum permissible speed for DC motors is primarily limited by the commutation system. A further reason for limiting the speed is the rotor's residual mechanical imbalance which shortens the service life of the bearings. The required peak speed n_p corresponding to the motor has to be smaller than or equal to the maximum permissible speed N_m^{max} of the motor

$$n_p \leq N_m^{max} \tag{3.7}$$

where $n_p = \max\{|2\pi\dot{\theta}(t) \cdot \rho|\}$.

The inequalities (3.5) to (3.7) represent the constraints that must be fulfilled by any motor in the drive train.

3.4.2 Gearbox selection criteria

In the selection of gearboxes, the following three constraints are considered:

Rated output torque limit. It is recommended by the Harmonic Drive gearbox manufacturer to use the RMC value for calculating rated torque [50]. The RMC value is a measure of the accumulated fatigue on a structural component and reflects typical endurance curves of steel and aluminium [51]. It is therefore relevant to gearbox lifetime, and this criterion has also been used in robotic applications [52]. With this criterion, a constraint is derived as

$$\tau_{rmc} \leq T_g \tag{3.8}$$

where $\tau_{rmc} = \sqrt[3]{\frac{1}{\Delta t}\int_0^{\Delta t} \tau^3(t) dt}$, with $\tau(t)$ being the required torque from the gearbox output. T_g is the limit for rated torque of the gearbox.

Maximum output torque limit. The required peak torque τ_g with respect to the output side has to be smaller than or equal to the allowable peak torque T_g^{max} of the harmonic drive

$$\tau_g \leq T_g^{max} \tag{3.9}$$

where $\tau_g = \max\{|\tau(t)|\}$.

Maximum permissible input speed limit. The required maximum input peak speed n_{in} has to be smaller than or equal to the maximum permissible input speed N_g^{max} of a gearbox

$$n_{in} \leq N_g^{max} \tag{3.10}$$

where $n_{in} = \max\{|\dot{\theta}(t) \cdot \rho|\}$.

The inequalities (3.8) to (3.10) represent the constraints that must be fulfilled by any gearbox in the drive train.

Although the inequalities (3.5) to (3.10) are derived specifically from the selection criteria for the motors and gearboxes considered in this paper, they are quite general, and would be recognized in any selection procedure for motors and gearboxes suitable for robotic arm design.

3.4.3 Objective function formulation

The objective of the optimization is to minimize the mass of the robotic arm. In this formulation, we minimize only the mass of the power transmission, while the mass of the arm structures (m_{arm}) remains constant. Therefore, the optimization task is to find the lightest combination of motor and gearbox for all five dof that fulfill all constraints associated with the motors and gearboxes. The objective function, $f(\mathbf{x})$, is defined as the sum of the mass of the motors and gears, as shown in Equation (3.11a).

$$\min_{\mathbf{x}} \quad f(\mathbf{x}) = \sum_{i=1}^{5} \{m_m(\mathbf{u}_m) + m_g(\mathbf{u}_g)\}_i \tag{3.11a}$$
$$\mathbf{x} = [\mathbf{u}_m, \mathbf{u}_g]$$

S.T.

$$T_{m,i} \geq \sqrt{\frac{1}{\Delta t} \int_0^{\Delta t} \left\{ \left(J_m(\mathbf{x}) + J_g(\mathbf{x})\right) \ddot{\theta}(t)\rho + \frac{\tau(t,\mathbf{x})}{\rho\eta_g} \right\}_i^2 \cdot dt} \tag{3.11b}$$

$$T_{m,i}^{max} \geq \max \left\{ \left| (J_m(\mathbf{x}) + J_g(\mathbf{x}))\ddot{\theta}(t)\rho + \frac{\tau(t,\mathbf{x})}{\rho\eta_g} \right| \right\}_i \tag{3.11c}$$

$$N_{m,i}^{max} \geq \max\left\{|2\pi\dot{\theta}(t)\cdot\rho|\right\}_i \tag{3.11d}$$

$$T_{g,i} \geq \sqrt[3]{\frac{1}{\Delta t}\int_0^{\Delta t}\tau_i^3(t,\mathbf{x})\cdot dt} \tag{3.11e}$$

$$T_{g,i}^{max} \geq \max\left\{|\tau(t,\mathbf{x})|\right\}_i \tag{3.11f}$$

$$N_{g,i}^{max} \geq \max\left\{|\dot{\theta}(t)\cdot\rho|\right\}_i \tag{3.11g}$$

where design variables in $\mathbf{x}$ includes the index numbers of motors $\mathbf{u}_m = [u_{m1},\ldots,u_{m5}]$ and gearboxes $\mathbf{u}_g = [u_{g1},\ldots,u_{g5}]$, relative to databases containing commercially available components.

So far, we have formulated the design problem as a discrete optimization problem, which can be solved by commercial available codes. We select a non-gradient method called Complex for this purpose. The implementation is outlined in the next section.

3.5 Procedure of optimization

The optimization method is developed as a Matlab and MSC.ADAMS co-simulation platform. The optimization algorithm is based on the Complex method, which is briefly discussed.

3.5.1 Optimization by Complex

The Complex method is a non-gradient based optimization method, first presented by Box [31].

In the Complex method, several possible designs (design population) are manipulated. The method is based on a feasible domain, containing a design population as a set of design points. The number of design points has to be greater than the number of independent design variables. The starting design points (initial population) are randomly generated, and evaluated through the objective function to check performance and constraint violation. Among all populations, the set of design variables having the minimal objective function is denoted as the best point $\mathbf{x}_b$, while the one having the maximal objective function is denoted as the worst point $\mathbf{x}_w$. Their corresponding values of objective function are noted as the best and worst values. The centroid point is calculated as

$$\mathbf{x}_c = \frac{1}{m-1}\sum_{i=1}^{m}\mathbf{x}_i, \quad \mathbf{x}_i \neq \mathbf{x}_b \tag{3.12}$$

$$\mathbf{x}_i = [x_1, x_2, \ldots, x_n], \quad m > n \tag{3.13}$$

The main idea of the Complex method is to replace the worst point by a new and better point. The new point is found by the reflection of the worst point through the centroid with a reflection coefficient α, yielding the following expression for the new design point

$$\mathbf{x}_{cand} = \mathbf{x}_c + \alpha(\mathbf{x}_c - \mathbf{x}_w) \tag{3.14}$$

The coefficient $\alpha = 1.3$ is used in this study, as recommended in [31]. The candidate point $\mathbf{x}_{cand}$ is checked through explicit and implicit constraints. When it conforms to the constraints, $\mathbf{x}_{cand}$ replaces $\mathbf{x}_w$. This method cannot handle the situation when the centroid is trapped in a local minimum. Therefore, the method has been modified such that the point moves toward the best point if it continues to be the worst one. To avoid the collapse of the algorithm, a random value is also added to the new point. The modified method to calculate the reflection point is given as

$$\mathbf{x}_{cand}^{new} = \frac{1}{2}\left(\mathbf{x}_{cand}^{old} + \varepsilon\mathbf{x}_c + (1-\varepsilon)\mathbf{x}_b\right) + (\mathbf{x}_c - \mathbf{x}_b)(1-\varepsilon)(2k-1) \tag{3.15}$$

where k is a random number varying in the interval $[0, 1]$, with

$$\varepsilon = \left(\frac{n_r}{n_r + k_r - 1}\right)^{\frac{n_r + k_r - 1}{n_r}} \tag{3.16}$$

Here k_r is the number of times the same point has repeatedly been identified as the worst point, and n_r is a tuning parameter which is set to 4. The convergence criterion of the Complex method in this work is the difference between the best and worst objective function values is less than a user defined tolerance.

3.5.2 Dynamics model with MSC.ADAMS

The drive requirements of the whole robotic arm system are determined from inverse kinematic and dynamic analysis within MSC.ADAMS. The inverse kinematic and dynamic analysis is developed as a simulation package, which will be called by the optimization program. To this end, the mass of motors and gearboxes are parameterized, while the trajectory of the robotic arm is prescribed. For each variation of motors and gearboxes, the required motor torques are accurately calculated. The mass of distribution is updated during the optimization procedure.

The inverse kinematic and dynamic analysis of the robotic arm in ADAMS follows a so-called master-slave approach, as shown in Fig. 3.5. The basic concept of this approach is that we make two models of the robotic arm in ADAMS, a master model and a slave one. In the master model, the inverse kinematic analysis is executed to record the joint motions corresponding to the prescribed end-effector trajectory. In the slave model, the joint motion data is imported and imposed on the joints, and payload is also attached to the end-effector. Then the inverse dynamic calculation is performed to solve the required joint torques for actuating the robotic arm.

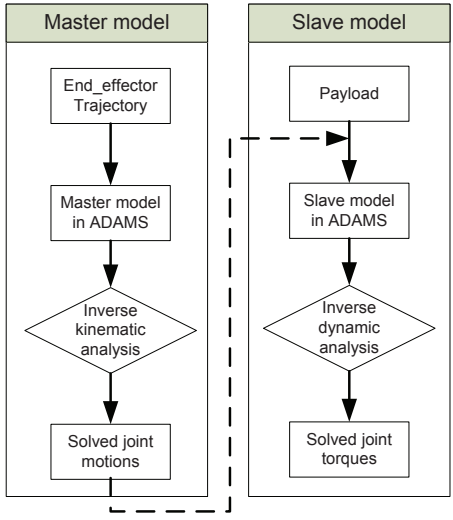


Figure 3.5: The procedure of inverse kinematic and dynamic analysis.

In the master-slave approach, we can define different trajectories and payloads for the robotic arm model, which makes the model more flexible for different simulation conditions. This approach can be applicable to other serial and parallel robot systems.

3.5.3 Matlab-ADAMS co-simulation platform

The design optimization is mainly concerned of two tasks: the optimization routine and creation of a parametric dynamic simulation model. Both tasks can be performed on a Matlab-ADAMS co-simulation platform developed in this work. As shown in Fig. 3.6, the platform works with two modules. The ADAMS module is used to simulate the inverse kinematics and dynamics of the robotic arm. The Matlab module implements the Complex method

to call the ADAMS simulation in batch mode.

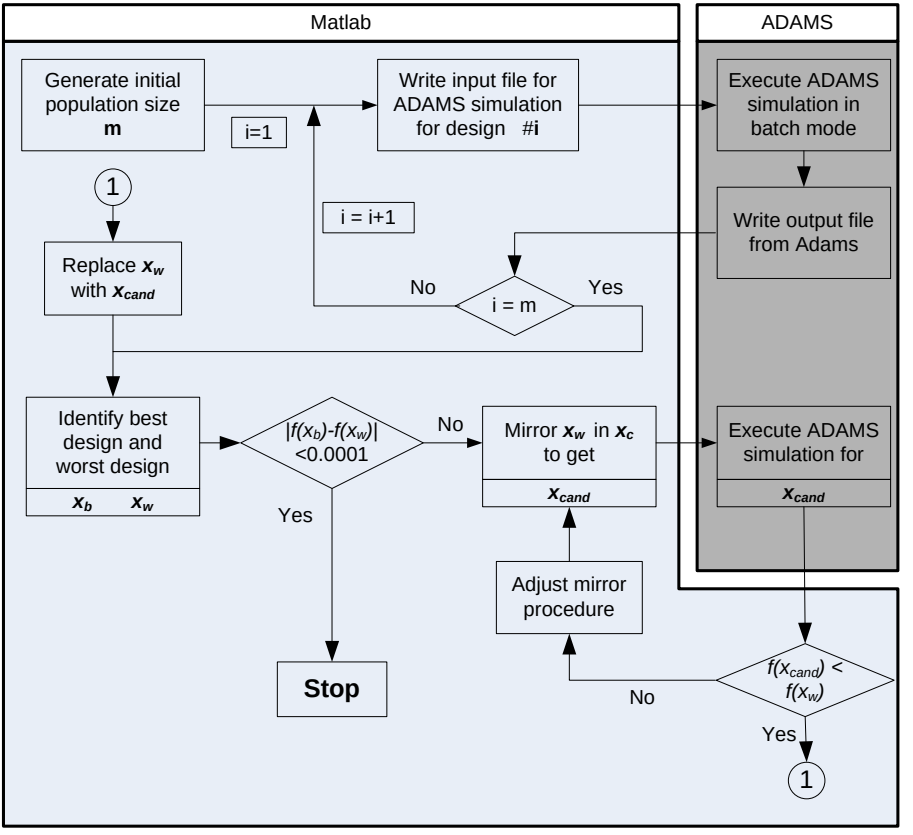
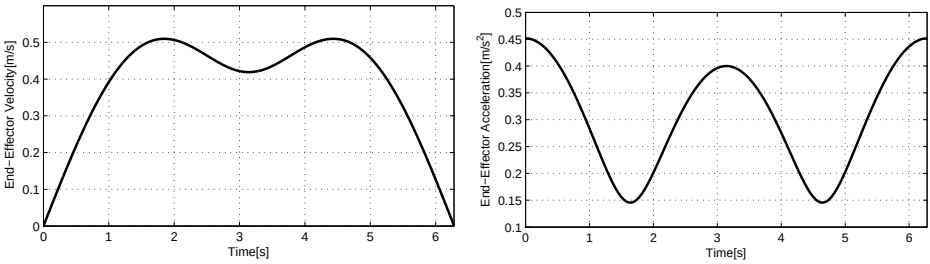


Figure 3.6: Diagram of the optimization routine in the co-simulation platform.

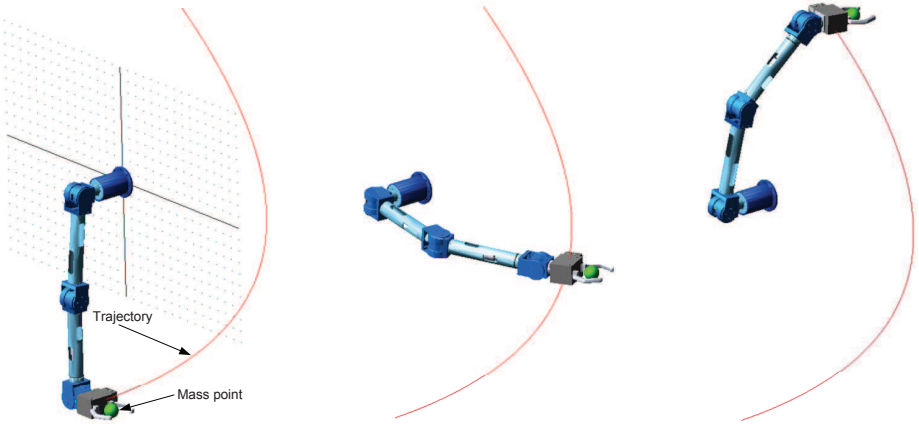
3.6 An example of design optimization

Design optimization was conducted on the 5-dof light-weight robotic arm. The link lengths of the robotic arm are fixed. The trajectory of the end-effector in the base coordinate system is defined as $X_{ef}(t) = 50 + 400(1 - \cos(t))$, $Y_{ef}(t) = -1000 + 800(1 - \cos(t/2))$, and $Z_{ef}(t) = 280 + 250(\cos(t/2) - 1)$, all with unit of mm. The corresponding velocity and acceleration profiles of the trajectory are depicted in Fig. 3.7. The Euler angles for the end-effector are given as $[\sin(t/180), 0, 0]$, which implies the end-effector remains horizontal during the prescribed motion. The motion of the end-effector is illustrated in Fig. 3.8.



(a) Velocity profile (b) Acceleration profile

Figure 3.7: Velocity and acceleration of the end-effector trajectory.



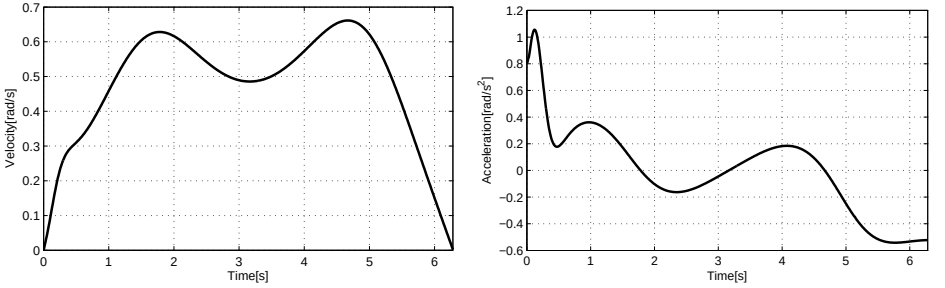
(a) (b) (c)

Figure 3.8: Illustration of a prescribed end-effector motion.

The payload is defined as a point of mass in ADAMS and weights $2kg$. On the other hand, the mass of motors and gearbox are determined from their indices selected. The solved motions of each joint, as shown in Fig. 3.9 for Joint 1, are imported into ADAMS to generate arm dynamics.

Ten candidate motors from the Maxon Motor catalogue are arranged ascendingly with respect to the mass of motor, as shown in Table 3.2.

The gearboxes used in the robotic arm are selected from Harmonic Drive CPU units, as listed in Table 3.3. For the Harmonic Drive gearboxes, the efficiency is a function of operation speed. In this paper, the gear efficiency is set as $\eta_g = 0.85$ for all gearboxes, for simplicity.



(a) Velocity profile (b) Acceleration profile

Figure 3.9: Velocity and acceleration of the 1st joint.

Table 3.2: Candidate motor data from Maxon Motor [53].

Index No.	Maxon Motor	T_m [Nm]	T_m^{max} [Nm]	N_m^{max} [rpm]	J_m $[g \cdot cm^2]$	m_m [kg]
1	EC 45 flat	0.0843	0.822	10000	135	0.11
2	RE 25	0.0284	0.28	14000	10.5	0.13
3	RE 26	0.0321	0.227	14000	12.1	0.15
4	EC-i 40	0.0667	1.81	15000	24.2	0.21
5	RE 30	0.0882	1.02	12000	34.5	0.238
6	EC 32	0.0426	0.353	25000	20	0.27
7	RE 35	0.0965	0.967	12000	67.4	0.34
8	RE 36	0.0795	0.785	12000	67.2	0.35
9	EC 40	0.127	0.94	18000	85	0.39
10	RE 40	0.184	2.5	12000	138	0.48

Table 3.3: Candidate gearbox data from Harmonic Drive [54].

Index No.	CPU Unit Size	Ratio	T_g [Nm]	T_g^{max} [Nm]	N_g^{max} [rpm]	J_g $[kg \cdot m^2]$	m_g [kg]
1	14	100	11	54	8500	0.033×10^{-4}	0.54
2	17	100	39	110	7300	0.079×10^{-4}	0.79
3	20	100	49	147	6500	0.193×10^{-4}	1.3
4	25	100	108	284	5600	0.413×10^{-4}	1.95

In order to simplify the process of selecting motors and gearboxes, the gear ratio of each joint is fixed as $\boldsymbol{\rho} = [344, 444, 100, 51, 100]$, orderly from Joints 1 to 5. The gear ratio is based on previous investigation of joint torques. The

ratios ρ_1 and ρ_2 are combinations of two gearboxes, a planetary gearhead and a Harmonic Drive unit, to achieve high gear ratio. For simplicity, the mass of the planetary gearhead is fixed, while only the mass of the Harmonic Drive gearbox is manipulated.

The gearbox is selected for each joint, associated with the selection of motor. The Harmonic Drive CPU unit is used in all joints except Joint 4, due to the joint structure consideration. A planetary gearhead is used in Joint 4, so $u_{g4} = 0$.

3.6.1 Optimization results

An optimized design of motor and gearbox for the robotic arm was found, as listed in Table 3.4. The optimized weight of the robotic arm is 10.2 kg, with a reduction of 38% corresponding to the initial combination of motors and gearboxes.

Table 3.4: Optimization results for minimization of weight.

Joint	Initial		Optimized	
	Motor	Gearbox	Motor	Gearbox
1	RE 40	CPU 17	EC 40	CPU 17
2	RE 35	CPU 17	EC 45 flat	CPU 17
3	RE 35	CPU 17	EC-i 40	CPU 14
4	RE 35	Gearhead	EC 45 flat	Gearhead
5	RE 35	CPU 17	EC 45 flat	CPU 14
Arm weight[kg]	16.7		10.2	

The convergence of the objective function is depicted in Fig. 3.10, where both the best and the worst objective function values from the Complex algorithm are displayed. The solution to the optimal result is achieved after 3160 iterations with a population size of 140. In this work, the tolerance of convergence criterion is set to 0.0001. It is noted from Fig. 3.10 that at the 1500th iteration, the difference between the best and the worst $f(\mathbf{x})$ values is 0.03, which means the convergence criterion is not met at that point, even though two values appear to be very closer.

Figure 3.11(a) illustrates the convergence of the motor design variables. Note that only convergence curves for Joints 1 and 5 are displayed for clarity. The convergence of the gearbox design variables is depicted in Fig. 3.11(b). Comparing the convergence rate for the motor and gearbox design variables,

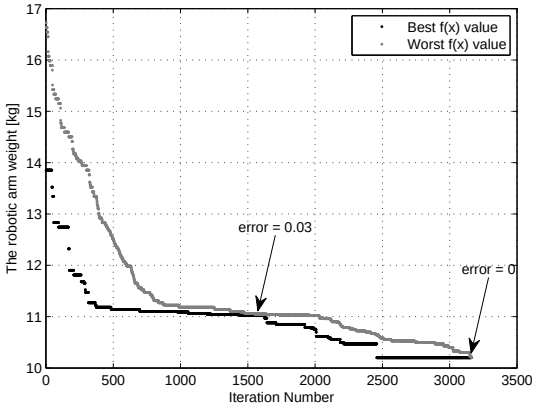
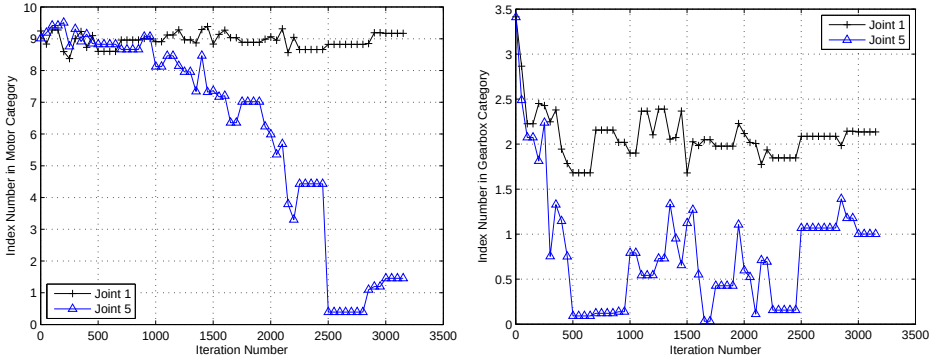


Figure 3.10: Convergence plot for the weight of the robotic arm.

the gearbox design variables converge faster than the motor design variables. A possible explanation is the mass difference between the Harmonic Drive units is larger than that between the motors.

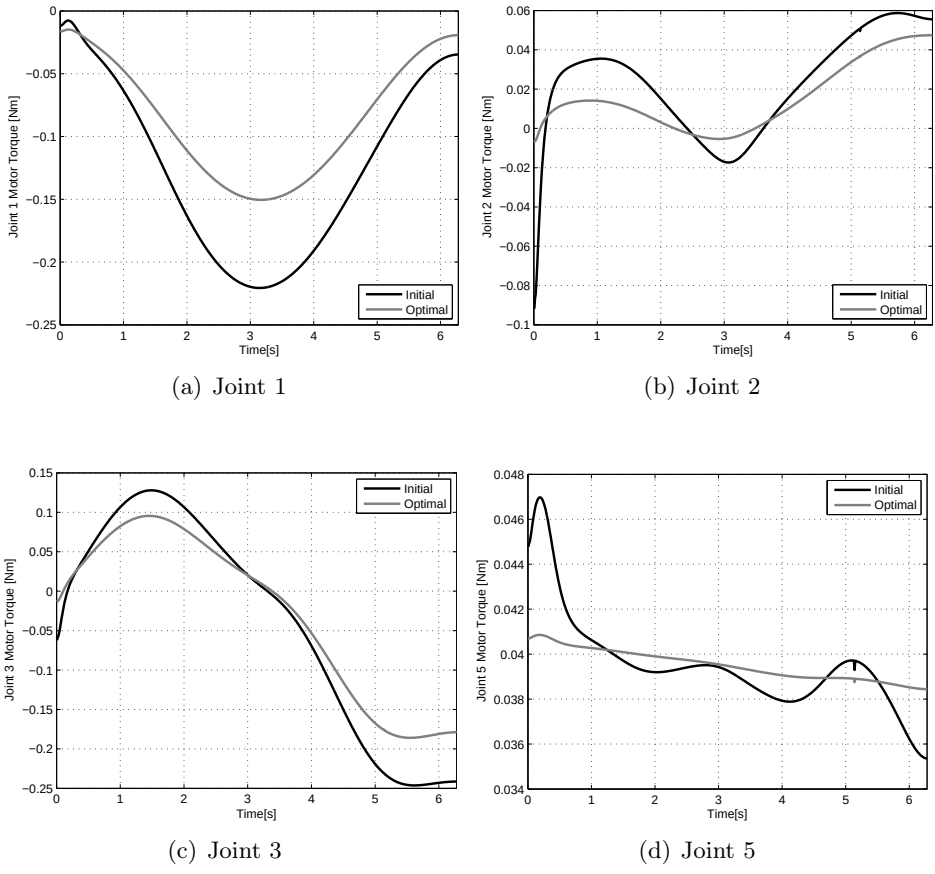


(a) Motors (b) Gearboxes

Figure 3.11: Convergence plots for design variables.

Based on the optimization results, the motor torques are obtained for Joints 1, 2, 3 and 5, as shown in Fig. 3.12, where torques for initial designs are also displayed for comparison. It is seen that the optimal design reduces the peak torque by 31.8% reduction for Joint 1 and by 40% for Joint 2.

To verify the accuracy of the solved joint torques from the co-simulation platform, another program was developed for simulation with Matlab only.

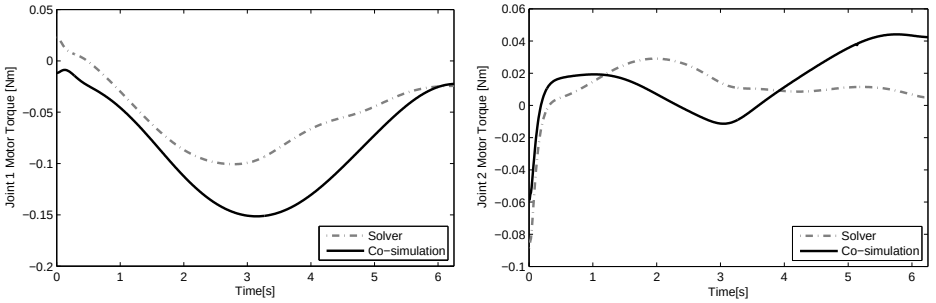


(a) Joint 1 (b) Joint 2

(c) Joint 3 (d) Joint 5

Figure 3.12: Motor torques for initial and optimal drive train combinations.

The joint torques obtained with the two methods are shown in Fig. 3.13 for the the same robot trajectory. Higher torques calculated by the co-simulation platform were observed for both Joints 1 and 2. The reason is that ADAMS in the co-simulation platform can calculate torques with more precise mass distribution, while the Matlab solver calculates the mass matrices using simple and regular geometry of links. The difference in mass matrices is demonstrated in Eq. (3.17) for Link 2, where $\mathbf{I}_2'$ is calculated by the Matlab solver and $\mathbf{I}_2$ is by ADAMS.



(a) Joint 1 (b) Joint 2

Figure 3.13: Comparisons of joint torques solved by the co-simulation platform and a Matlab solver.

$$\mathbf{I}_2' = \begin{bmatrix} 0.0076 & 0 & 0 \\ 0 & 0.0076 & 0 \\ 0 & 0 & 0.0045 \end{bmatrix} [kg{\cdot}m^2], \quad \mathbf{I}_2 = \begin{bmatrix} 0.008 & -0.00032 & -0.00008 \\ -0.00032 & 0.0069 & 0.00037 \\ -0.00008 & 0.00037 & 0.0049 \end{bmatrix} [kg{\cdot}m^2] \tag{3.17}$$

3.6.2 Design variables programming

The design points in the Complex method are usually continuous. But the design variables $u_{m,i}$ and $u_{g,i}$ have to be integers, since they are the index numbers from the databases of motors and gearboxes. Two ways of dealing with the design variables are investigated in order to confirm a more efficient one. One way is called *Rounded Design Variable (RDV)*, the other one called *Linear Design Variable (LDV)*.

Rounded Design Variable (RDV) For the RDV method, a rounding function is introduced to transfer the design variables into integers. The rounding function is given as

$$x_{DV} = round(x) = \begin{cases} x_{int}; & if\ x_{int} \le x < x_{int} + 0.5 \\ x_{int} + 1; & if\ x_{int} + 0.5 \le x < x_{int} + 1 \end{cases} \tag{3.18}$$

where x is the design variable manipulated by the Complex method, x_{int} is the integral part of the number x, and x_{DV} is the rounded design variable. x_{DV} is used to update the mass of motors and gearboxes in inverse dynamic analysis, as well as the allowable torque and speed values used to examine constraint violations.

Linear Design Variable (LDV) For the LDV method, the mass between two adjacent motors (or gearboxes) in the category is linearized by the function

$$m(x) = m(x_{int}) + (x - x_{int}) \cdot [m(x_{int} + 1) - m(x_{int})] \tag{3.19}$$

where $m(x_{int})$ is the mass of the component (motors from Table 3.2 and gearboxes from Table 3.3) corresponding to the index number x_{int}, $m(x)$ is the mass to be updated for the component in inverse dynamic analysis.

Comparison has been conducted upon the RDV and LDV methods of dealing with the integer design variables, as listed in Table 3.5. In general, the LDV method yields better results at the cost of more iterations and objective function evaluations. In the above example, the RDV method has been preferred, however, the choice of method must in general be a compromise between accuracy and optimization time.

Table 3.5: The comparisons of RDV and LDV.

Population	RDV		LDV	
	$f(\mathbf{x})$	Iteration	$f(\mathbf{x})$	Iteration
20	11.142	213	10.633	1499
40	10.952	927	10.703	2607
60	10.458	1444	10.414	6303
80	10.402	3026	10.122	7306

3.7 Conclusions

A method was developed for the optimum design of robotic drive trains. The selection of motors and gears is formulated as a discrete optimization problem, which is solved by a non-gradient optimization algorithm. Constraints were formulated by considering both motor and gearbox characteristics and robotic arm dynamics. The proposed method is able to reach a design with lower mass for a given set of driving components. A co-simulation platform consisting of a MSC.ADAMS dynamics model and an optimization algorithm implemented in Matlab code was developed, which enables design optimization based on dynamics of an embodiment existing in CAD systems. Such a platform not only yields accurate dynamic calculation for drive train optimization, but also leads to a possible integrated optimization for both arm structures and drive trains.

Appendix 1

The transformation matrix in forward kinematics of the end-effector in fixed reference frame is given as:

$$ {}^{0}A_5 = \begin{bmatrix} \mathbf{R} & \mathbf{q} \\ 0 & 1 \end{bmatrix} \tag{3.20} $$

with the rotation matrix $\mathbf{R}$ and position vector $\mathbf{q}$ being given by

$$ \mathbf{R} = \begin{bmatrix} u_x & v_x & w_x \\ u_y & v_y & w_y \\ u_z & v_z & w_z \end{bmatrix}, \quad \mathbf{q} = \begin{bmatrix} q_x \\ q_y \\ q_z \end{bmatrix} \tag{3.21} $$

The forward kinematics of the robotic arm is solved as below:

$$
\begin{aligned}
u_x &= c\theta_5(s\theta_1 s\theta_4 + c\theta_1 c\theta_{23} c\theta_4) - c\theta_1 s\theta_{23} s\theta_5, \\
u_y &= -c\theta_5(c\theta_1 s\theta_4 - s\theta_1 c\theta_{23} c\theta_4) - s\theta_1 s\theta_{23} s\theta_5, \\
u_z &= c\theta_{23} s\theta_5 + s\theta_{23} c\theta_4 c\theta_5, \\
v_x &= s\theta_1 c\theta_4 - c\theta_1 c\theta_{23} s\theta_4, \\
v_y &= -s\theta_1 c\theta_{23} s\theta_4 - c\theta_1 c\theta_4, \\
v_z &= -s\theta_{23} s\theta_4, \\
w_x &= s\theta_5(s\theta_1 s\theta_4 + c\theta_1 c\theta_{23} c\theta_4) + c\theta_1 s\theta_{23} c\theta_5, \\
w_y &= -s\theta_5(c\theta_1 s\theta_4 - s\theta_1 c\theta_{23} c\theta_4) + s\theta_1 s\theta_{23} c\theta_5, \\
w_z &= s\theta_{23} c\theta_4 s\theta_5 - c\theta_{23} c\theta_5, \\
q_x &= l_1 c\theta_1 c\theta_2 + l_2 c\theta_1 s\theta_{23} + d_1 \cdot (s\theta_1 c\theta_4 - c\theta_1 c\theta_{23} s\theta_4), \\
q_y &= l_1 s\theta_1 c\theta_2 + l_2 s\theta_1 s\theta_{23} - d_1 \cdot (c\theta_1 c\theta_4 + s\theta_1 c\theta_{23} s\theta_4), \\
q_z &= h_1 + l_1 s\theta_2 - l_2 c\theta_{23} - d_1 s\theta_{23} s\theta_4,
\end{aligned}
$$

where $c\theta_{23}$ stands for $\cos(\theta_2 + \theta_3)$, and $s\theta_{23}$ for $\sin(\theta_2 + \theta_3)$.

Appendix 2

The joint angles for a given pose in terms of $\mathbf{R}$ and $\mathbf{q}$ can be found through the inverse kinematics presented below.

Skipping details, a solution for θ_1 is found as

$$ \theta_1 = \arctan\left(\frac{p_y}{p_x}\right) \tag{3.22} $$

where $p_x = q_x - d_1 w_x$ and $p_y = q_y - d_1 w_y$. Equation (3.22) leads to two solutions of θ_1, i.e. $\theta_1 = \theta_1^*$ and $\theta_1 = \theta_1^* + \pi$, where $0 \leq \theta_1^* \leq \pi$. They

represent two branches of arm kinematics. The real value depends on the initial configuration.

The solution of θ_3 is given as

$$\theta_3 = \arctan\left(\pm\frac{\kappa_3 - \kappa_1}{\sqrt{{\kappa_2}^2 - (\kappa_3 - \kappa_1)^2}}\right); \quad \kappa_2^2 - (\kappa_3 - \kappa_1)^2 > 0 \tag{3.23}$$

where $\kappa_1 = l_1^2 + l_2^2$, $\kappa_2 = 2l_1l_2$, $\kappa_3 = p_x^2 + p_y^2 + (p_z - h_1)^2$, and $p_z = q_z - d_1w_z$.

Once θ_1 and θ_3 are known, θ_2 can be obtained as

$$\theta_2 = \arctan\frac{(\mu_2\eta_1 - \mu_1\eta_2)(\zeta_2\mu_1 - \zeta_1\mu_2)}{(\mu_2\zeta_1 - \mu_1\zeta_2)(\zeta_2\eta_1 - \zeta_1\eta_2)} \tag{3.24}$$

where

$$\mu_1 = l_1 + l_2s\theta_3,\ \zeta_1 = l_2c\theta_3,\ \eta_1 = p_xc\theta_1 + p_ys\theta_1;$$
$$\mu_2 = -l_2c\theta_3,\ \zeta_2 = l_1 + l_2s\theta_3,\ \eta_2 = p_z - h_1.$$

θ_5 takes the form of

$$\theta_5 = \arccos\left(w_xc\theta_1s\theta_{23} + w_ys\theta_1s\theta_{23} - w_zc\theta_{23}\right) \tag{3.25}$$

Assuming that $s\theta_5 \neq 0$, we can solve for θ_4 as follows.

$$\theta_4 = \arctan\left(s\theta_4, c\theta_4\right) \tag{3.26}$$

where

$$c\theta_4 = \frac{w_xc\theta_1c\theta_{23} + w_ys\theta_1c\theta_{23} + w_zs\theta_{23}}{s\theta_5}; \quad s\theta_4 = \frac{w_xs\theta_1 - w_yc\theta_1}{s\theta_5}.$$

4

Article II

Integrated Dimensional and Drive-Train Design Optimization of a Light-Weight Anthropomorphic Arm

The paper has been published as:

Lelai Zhou, Shaoping Bai, and Michael R. Hansen, "Integrated Dimensional and Drive-Train Design Optimization of a Light-Weight Anthropomorphic Arm", *Robotics and Autonomous Systems*, vol. 60, no. 1, pp:113-122, 2012. doi:10.1016/j.robot.2011.09.004

Abstract

An approach to minimize the mass of robotic manipulators is developed by integrated dimensional and drive-train optimization. The method addresses the influences of dimensions and characteristics of drive-trains in the design optimization. Constraints are formulated on the basis of kinematic performance and dynamic requirements, whereas the main objective is to minimize the total mass. Case studies are included to demonstrate the application of the optimization method in the design of assistive robots.

4.1 Introduction

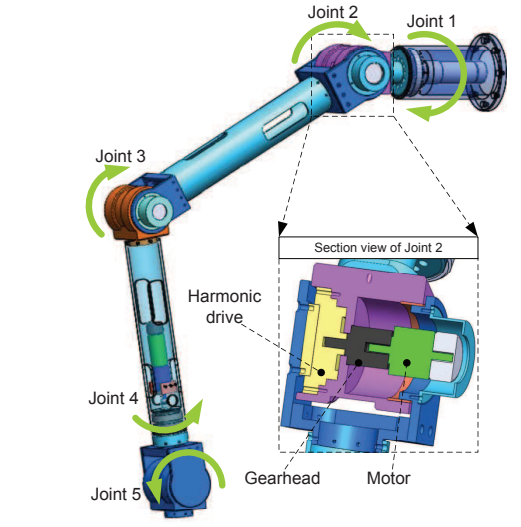
Assistive robotic systems consisting of a robot arm mounting on a moving platform become increasingly important in assisting the handicapped and elderly people. Typical systems reported include FRIEND-I and FRIEND-II [1] from University of Bremen, KARES II [2], and RAPTOR [3]. For such systems, light-weight arms with high payload capacity are desirable from the point of view of both safety and energy efficiency. Specifically, it is desired that injuries caused by collision between robotic components and human beings are minimized in case of accidents.

The design of a light and strong robotic arm faces many challenges, varying from power supply, actuators, power transmission and structural parts. New technologies have been developed in connection with some novel robotic arms. DLR's robotics lab designed a 7-dof (degrees of freedom) torque-controlled light-weight robotic arm with a payload-to-weight ratio of 1 and a payload mass of $14kg$ using customer designed drive-trains and carbon fiber structures [4]. Jardon et al. [6] built a 5-dof self-containing light-weight manipulator with a portable concept from wheelchair to docking stations in the room. Ananiev et al. [9] designed a 6-dof light-weight reconfigurable robotic arm mounted on a mobile platform. The 7-dof light-weight Whole Arm Manipulator (WAM) developed by Barrett Technology takes advantage of differential mechanisms driven by cables and cylinder transmissions to reduce the effective arm inertia [8].

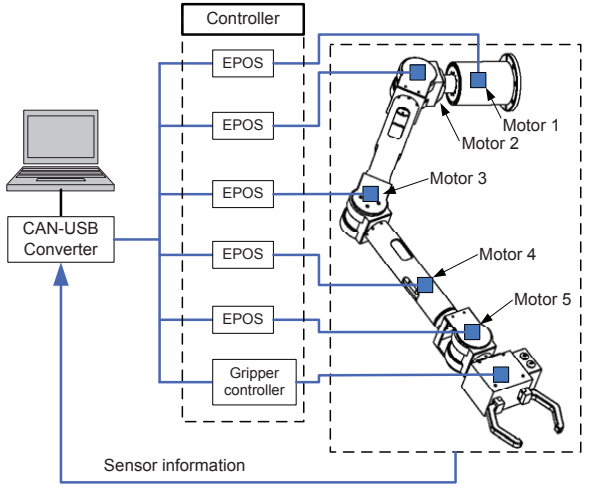
It is realized that the mass of an arm comes mainly from the structural parts and the drive-trains. A majority of research work in design optimization is related to the drive-train design. An early attempt on drive-train design optimization can be found in [18], in which Chedmail and Gautier proposed a method for the optimum selection of robot actuators to minimize the

total mass of all the actuators. Pettersson and Ölvander [19] reported a method of design optimization, in which the drive-train of two joints was optimized for an industrial manipulator. A simulation environment called Modelica with robot optimization facilities was presented in [20], where the parameters of a controller were tuned by a multi-criteria parameter optimization method to improve the system dynamics. A drive-train optimization for robot designing was recently reported in [55]. The method is able to optimally select combinations of motors and gearboxes from a catalogue of commercially available components for each dof of a robot arm. On the other hand, dimensional optimization was also studied for the improvement of robotic performance, either kinematic or dynamic one. An optimum robot design method based on a specified task was proposed [25], in which dimensions were optimized based on dynamic analysis. A method to find optimal manipulator parameters using evolutionary optimization algorithms was presented in [24]. Methods based on optimal dynamic performance were reported in [21, 22, 26, 27], among others, in which the influence of dimensions of robotic manipulators was rarely considered. It can be noticed that for most researches, dimensional and drive-train optimizations were conducted separately. An integrated approach is desired in order to fully utilize the potential of applying optimization techniques to robot design.

This paper reports an integrated dimensional and drive-train optimization method for the design of robotic manipulators. Our interest is to include the dimensions of a robotic arm as variables in the design optimization, in addition to the parameters of the drive-trains. The inclusion of the dimensions in the optimization will allow us to account for their influence on the kinematic and dynamic performances, both being major concerns in the robot design. The work in this paper was carried out for a light-weight robotic arm of five degrees of freedom (dof), with two dof at the shoulder, one at the elbow, and two at the wrist, as shown in Fig. 4.1(a). This is a human-like arm design, which is to be mounted on an electric wheelchair to assist disabled people in simple manipulations like picking, placing, door opening, etc. For this purpose, a gripper is employed at the end of the arm, as demonstrated in Fig. 4.1(b). A design with minimal mass can make the robot intrinsically safe in assistive manipulations.



(a)



(b)

Figure 4.1: (a) CAD rendering of a 5-dof light-weight anthropomorphic arm, (b) control system of the robotic arm.

4.2 The anthropomorphic arm and modeling

The 5-dof robotic arm adopts a modular approach. As shown in Fig. 4.1(a), CPU series gearboxes of Harmonic Drive™ are used as transmission elements

and, simultaneously, as the mechanical joints, for different dofs. To increase the torque capabilities of Joints 1, 2 and 3, a second stage of gearhead is used between Harmonic Drive and the motor. The geared motors and Harmonic Drive gearboxes are mounted inside the joint housings, while the axes of rotation coincide with the joint axes, as illustrated in Fig. 4.1(a). The arm joints are driven by electrical motors, chosen among Maxon™ DC motors. The motors are equipped with encoders having 1000 counts per turn.

CANopen (Controller Area Network) bus is adopted for the communications between motors and controllers. As shown in Fig. 4.1(b), the motors are controlled by EPOS controllers, which are selected from Maxon. The gripper, selected from SOMMER™ Automatic, is controlled by its customized controller. CAN runs a two-wire differential serial communication protocol, the CANopen protocol, for real-time control. CANopen protocol uses the CAN Physical Layer as defined by the CAN in Automation (CiA) standard 'DS-301 Version 4.02'. The communications between CANopen bus and the PC are accomplished by a CAN-USB interface.

4.2.1 Kinematics

Following the Denavit-Hartenberg (D-H) convention [38], Cartesian coordinate systems are established for each link of the robotic arm, as shown in Figure 4.2. D-H parameters are defined as listed in Table 4.1.

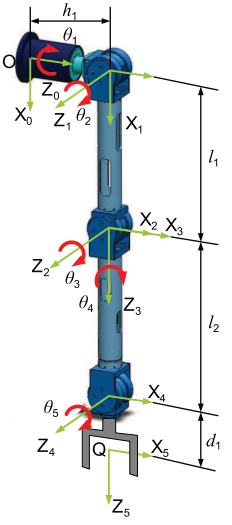


Figure 4.2: Robotic arm coordinate systems.

Table 4.1: D-H parameters of the robotic arm.

Joint i	α_i	a_i	d_i	θ_i
1	$\pi/2$	0	h_1	θ_1
2	0	l_1	0	θ_2
3	$\pi/2$	0	0	θ_3
4	$-\pi/2$	0	l_2	θ_4
5	$\pi/2$	0	d_1	θ_5

The transformation matrix in forward kinematics of the end-effector in fixed reference frame is given as

$$ {}^{0}A_5 = \begin{bmatrix} \mathbf{R} & \mathbf{q} \\ 0 & 1 \end{bmatrix} \tag{4.1} $$

with the rotation matrix $\mathbf{R}$ and position vector $\mathbf{q}$ given by

$$ \mathbf{R} = \begin{bmatrix} u_x & v_x & w_x \\ u_y & v_y & w_y \\ u_z & v_z & w_z \end{bmatrix}, \quad \mathbf{q} = \begin{bmatrix} q_x \\ q_y \\ q_z \end{bmatrix} \tag{4.2} $$

The joint angles for a given pose in terms of $\mathbf{R}$ and $\mathbf{q}$ can be found through the inverse kinematics. For simplicity, only solutions of the five joint angles are presented, which are

$$ \theta_1 = \arctan\left(\frac{p_y}{p_x}\right),\ 0 \le \theta_1 \le \pi \tag{4.3a} $$

$$ \theta_3 = \arctan\left(s\theta_3, c\theta_3\right) \tag{4.3b} $$

where

$$ p_x = q_x - d_1 w_x,\ p_y = q_y - d_1 w_y,\ p_z = q_z - d_1 w_z $$

$$ \kappa_1 = l_1^2 + l_2^2,\ \kappa_2 = 2 l_1 l_2,\ \kappa_2^2 - (\kappa_3 - \kappa_1)^2 > 0 $$

$$ \kappa_3 = p_x^2 + p_y^2 + (p_z - h_1)^2 $$

$$ s\theta_3 = (\kappa_3 - \kappa_1)/\kappa_2,\ c\theta_3 = \sqrt{1 - ((\kappa_3 - \kappa_1)/\kappa_2)^2} $$

Upon solved θ_1 and θ_3, other joint angles are obtained as

$$\theta_2 = \arctan \frac{(\mu_2\eta_1 - \mu_1\eta_2)(\zeta_2\mu_1 - \zeta_1\mu_2)}{(\mu_2\zeta_1 - \mu_1\zeta_2)(\zeta_2\eta_1 - \zeta_1\eta_2)},\ 0 \leq \theta_2 \leq \pi \tag{4.3c}$$

$$\theta_5 = \arccos\left(w_x c\theta_1 s\theta_{23} + w_y s\theta_1 s\theta_{23} - w_z c\theta_{23}\right),\ 0 < \theta_5 < \pi \tag{4.3d}$$

$$\theta_4 = \arctan\left(s\theta_4, c\theta_4\right) \tag{4.3e}$$

with

$$\mu_1 = l_1 + l_2 s\theta_3,\ \zeta_1 = l_2 c\theta_3,\ \eta_1 = p_x c\theta_1 + p_y s\theta_1$$

$$\mu_2 = -l_2 c\theta_3,\ \zeta_2 = l_1 + l_2 s\theta_3,\ \eta_2 = p_z - h_1$$

$$c\theta_4 = \frac{w_x c\theta_1 c\theta_{23} + w_y s\theta_1 c\theta_{23} + w_z s\theta_{23}}{s\theta_5}$$

$$s\theta_4 = \frac{w_x s\theta_1 - w_y c\theta_1}{s\theta_5},\ s\theta_5 \neq 0$$

where c and s stand for harmonic functions cosine and sine, respectively. Moreover $\theta_{23} \equiv \theta_2 + \theta_3$.

4.2.2 Jacobian matrix

The joint angular velocity can be calculated with the Jacobian matrix

$$\dot{\boldsymbol{\theta}} = \mathbf{J}^{-1}\mathbf{v}_{ef} \tag{4.4}$$

where $\dot{\boldsymbol{\theta}} = [\dot{\theta}_1, \dot{\theta}_2, \ldots, \dot{\theta}_n]^T$ denotes an n-dimensional (n denotes the number of dof) vector of the joint angular velocities, $\mathbf{J}$ is the Jacobian of the robotic arm, and $\mathbf{v}_{ef}$ the velocity of the end-effector.

For a revolute joint, the Jacobian matrix can be calculated by [39]

$$\mathbf{J} = [\mathbf{j}_1, \mathbf{j}_2, \ldots, \mathbf{j}_n],\ \mathbf{j}_i = \begin{bmatrix} \mathbf{z}_{i-1} \times \mathbf{p}_{i-1} \\ \mathbf{z}_{i-1} \end{bmatrix} \tag{4.5}$$

where $\mathbf{z}_{i-1}$ and $\mathbf{p}_{i-1}$ are given by

$$\mathbf{z}_{i-1} = \mathbf{R}_{i-1}[0 \ \ 0 \ \ 1]^{\mathrm{T}},\ \mathbf{p}_{i-1} = \mathbf{R}_{i-1}\mathbf{q}_{i-1} + \mathbf{p}_i \tag{4.6}$$

where $\mathbf{q}_{i-1} = [a_i \cos\theta_i, a_i \sin\theta_i, d_i]^{\mathrm{T}}$, $\mathbf{R}_{i-1}$ denotes the rotation matrix from the reference coordinate system to the $(i-1)$th coordinate system. The local coordinates of the end-effector are defined as $\mathbf{p}_n = [0, 0, 0]^{\mathrm{T}}$. When the desired end-effector velocity $\mathbf{v}_{ef}$ is given, the joint angular velocity can be solved by Eq. (4.4).

4.2.3 Inverse dynamics

The integrated dimensional and drive-train optimization will make use of a dynamic model of the robotic arm for dynamic evaluations. The governing equation of the arm motion can be written as

$$\mathbf{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \mathbf{V}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \mathbf{G}(\boldsymbol{\theta}) = \boldsymbol{\tau} \tag{4.7}$$

where $\mathbf{M}$ is the mass matrix, $\mathbf{V}$ is the vector of Coriolis and centrifugal terms of the links, $\mathbf{G}$ is the vector of gravitational forces, $\boldsymbol{\tau}$ is the vector of joint torques, and $\boldsymbol{\theta}$ is the vector of joint angles.

The mass matrix $\mathbf{M}$ can be calculated as

$$\mathbf{M} = \sum_{i=1}^{n} (\mathbf{J}_{v,i}^{\mathrm{T}} m_i \mathbf{J}_{v,i} + \mathbf{J}_{\omega,i}^{\mathrm{T}} \mathbf{I}_i \mathbf{J}_{\omega,i}), \ \mathbf{J}_i = \begin{bmatrix} \mathbf{J}_{v,i} \\ \mathbf{J}_{\omega,i} \end{bmatrix} \tag{4.8}$$

where $\mathbf{J}_{v,i}$ and $\mathbf{J}_{\omega,i}$ are $3 \times n$ matrices. For revolute joint, the jth column vectors of $\mathbf{J}_{v,i}$ and $\mathbf{J}_{\omega,i}$ can be obtained by [39]

$$\mathbf{j}_{v,i}^{j} = \mathbf{z}_{j-1} \times \mathbf{p}_{c,i}^{j-1}, \ \mathbf{j}_{\omega,i}^{j} = \mathbf{z}_{j-1}, \quad \text{for} \quad j \leq i \tag{4.9a}$$

$$\mathbf{j}_{v,i}^{j} = \mathbf{j}_{\omega,i}^{j} = [0 \ \ 0 \ \ 0]^{T}, \quad \text{for} \quad i < j \leq n \tag{4.9b}$$

where $\mathbf{p}_{c,i}^{j-1}$ is a position vector defined from the origin of the $j-1$ link frame to the center of mass of link i and expressed in the base frame. Moreover, m_i and $\mathbf{I}_i$ are the mass and the inertia matrix of the link i. For each link, its mass is found as

$$m_i = m_{s,i} + m_{m,i} + m_{g,i} \tag{4.10}$$

where $m_{s,i}$ is the mass of the arm structure, which is proportional to the link length. $m_{m,i}$ and $m_{g,i}$ are the masses of motor and gearbox for the ith joint. Both m_i and $\mathbf{I}_i$ vary with the selections of motors and gearboxes, and the link lengths as well.

4.3 Integrated dimensional and drive-train optimization

The integrated dimensional and drive-train optimization is proposed to minimize the mass of the robotic arm with constraints on kinematic performance and the robotic dynamics. The selection of motor and gearbox for a drive-train is constrained through the dynamic requirements and the selecting criteria for motors and gearboxes. Since the geometric dimensions[1] influence

[1]Geometric dimensions in this work refer to the link lengths of the robotic arm's D-H parameters, namely, l_1 and l_2.

the robotic dynamics, and also determine the kinematic performance of the robotic manipulator; a constraint on the kinematic performance can be defined to account for the dimensions' influence.

With the objective to minimize the mass of the robotic arm, the optimization task is to find the lightest combination of motor and gearbox for all the joints and the optimal link lengths that fulfill all constraints associated with the kinematic performance, the motors and gearboxes. The optimization problem is defined as

$$\text{Minimize} \quad f(\mathbf{x}) = \sum_{i=1}^{n} \{m_m(\mathbf{u}_m) + m_g(\mathbf{u}_g)\}_i \tag{4.11a}$$

$$\mathbf{x} = [\mathbf{u}_m, \mathbf{u}_g, \mathbf{u}_d]$$

$$\text{subject to} \quad g_j(\mathbf{x}) \leq 0 \tag{4.11b}$$

where $f(\mathbf{x})$ is the total mass of the robotic arm, $g_j(\mathbf{x})$ is the set of inequality constraints. The array of design variable $\mathbf{x}$ includes the index numbers of motors $\mathbf{u}_m = [u_{m1}, \ldots, u_{mn}]$ and gearboxes $\mathbf{u}_g = [u_{g1}, \ldots, u_{gn}]$, relative to the database containing commercially available components, and an array of dimensional variables $\mathbf{u}_d$.

The set of constraints $g_j(\mathbf{x})$ includes the kinematic performance constraint, the constraints for motor selection, and that for gearbox selection, as described presently.

4.3.1 Global conditioning index

The kinematic performance is one of the major concerns in robot design. It is desirable for a robot to have a high kinematic performance, while the drive-drain being optimized. Several performance indices have been used in designs of robotic manipulators. Yoshikawa [56] proposed manipulability measure as a metric of kinematic performance. Gosselin and Angeles [57] developed a global conditioning index (GCI) for the kinematic optimization of manipulators. The condition number and dexterity indices of the manipulator have been adopted in optimum designs [58, 59, 60].

Among the performance indices mentioned, the manipulability measure is a local performance measure and valid at a certain position only [61]. In this work, we use a global performance index, the global conditioning index

(GCI). The GCI is defined over a workspace Ω as [57]

$$GCI = \frac{\int_{\Omega} \frac{1}{\kappa} \mathrm{d}W}{\int_{\Omega} \mathrm{d}W} \tag{4.12}$$

with the condition number κ given by

$$\kappa = \| \mathbf{J}(\boldsymbol{\theta}, \mathbf{u}_d) \| \| \mathbf{J}^{-1}(\boldsymbol{\theta}, \mathbf{u}_d) \| \tag{4.13}$$

where $\mathbf{J}(\boldsymbol{\theta}, \mathbf{u}_d)$ is the Jacobian matrix defined in Eq. (4.4). The Euclidean norm $\| \cdot \|$ of the matrix is defined as

$$\| \mathbf{J} \| = \sqrt{tr(\mathbf{J}^T \mathbf{N} \mathbf{J})} \tag{4.14}$$

with $\mathbf{N} = \frac{1}{n}\mathbf{I}$, where n is the dimension of the square matrix $\mathbf{J}$, and $\mathbf{I}$ is the $n \times n$ identity matrix.

In practice, the GCI of a robotic manipulator is calculated through a discrete approach as [62]

$$GCI = \frac{1}{W} \sum_{i=1}^{m} \frac{1}{\kappa_i} \Delta W_i \tag{4.15}$$

where W is the workspace volume, and m is the number of discrete points. In the case of equal-volumetric discretization, $\Delta W_i \equiv \Delta W$, Eq. (4.15) is simplified to

$$GCI = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{\kappa_i} \tag{4.16}$$

The GCI is dimension dependent, which means

$$GCI = GCI(\mathbf{u}_d) \tag{4.17}$$

To keep a high kinematic performance with selected link lengths in the integrated optimization, a constraint is given on the GCI

$$GCI(\mathbf{u}_d) \geq C_{min} \tag{4.18}$$

where C_{min} is a user-defined minimum acceptable GCI.

4.3.2 Drive-train modeling

A drive-train normally consists of a motor, a linkage and a gearbox for speed reduction. The drive-train model of a single robotic joint is shown in Fig. 4.3. For the Harmonic Drive gearbox, the gear efficiency varies relative to the output torque. The required motor torque for the ith joint is calculated by

$$\tau_{m,i} = \left\{ (J_m + J_g)\ddot{\theta}(t)\rho + \frac{\tau(t)}{\rho\eta_g} \right\}_i ; \qquad i = 1, \ldots, n, \tag{4.19}$$

where ρ_i is the gear ratio, $J_{m,i}$ is mass moment of inertia of the ith motor, $J_{g,i}$ is the equivalent mass moment of inertia of the ith gearbox, $\eta_{g,i}$ is the corresponding gear efficiency, and $\tau_i(t)$ is the load at the output link which can be solved by Eq. (4.7).

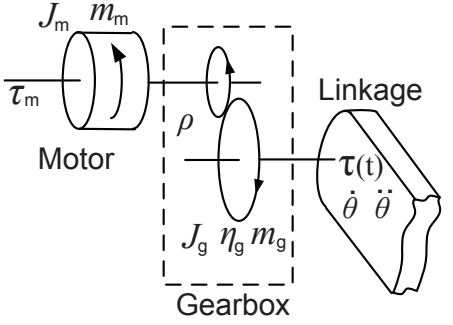


Figure 4.3: Schematic view of drive-train model for a single joint.

4.3.3 Motor selection criteria

In selecting motors, the following three constraints have to be satisfied, according to the motor selection criteria recommended by the manufacturer [53].

Nominal torque limit. The nominal torque is the so-called maximum continuous torque. The root mean square (RMS) value τ_{rms} of the required motor torque τ_m has to be smaller than or equal to the nominal torque of the motor T_m

$$\tau_{rms} \leq T_m \tag{4.20}$$

where $\tau_{rms} = \sqrt{\frac{1}{\Delta t}\int_0^{\Delta t} \tau_m^2 \mathrm{d}t}$, with Δt being the duration of a characteristic working cycle.

Stall torque limit. The stall torque is the peak torque of the motor. The required peak torque τ_p has to be smaller than or equal to the stall torque T_m^{max} of the motor

$$\tau_p \leq T_m^{max} \tag{4.21}$$

where $\tau_p = \max\{|\tau_m|\}$.

Maximum permissible speed limit. The maximum permissible speed for DC motors is primarily limited by the commutation system. A further reason for limiting the speed is the rotor's residual mechanical imbalance which shortens the service life of the bearings. The required peak speed n_p corresponding to the motor has to be smaller than or equal to the maximum permissible speed N_m^{max} of the motor

$$n_p \leq N_m^{max} \tag{4.22}$$

where $n_p = \max\{|2\pi\dot{\theta}(t) \cdot \rho|\}$.

4.3.4 Gearbox selection criteria

In the selection of gearboxes, the following three constraints are considered:

Rated output torque limit. It is recommended by the Harmonic Drive gearbox manufacturer to use the RMC value for calculating rated torque [50]. The RMC value is a measure of the accumulated fatigue on a structural component and reflects typical endurance curves of steel and aluminium [51]. It is therefore relevant to gearbox lifetime, and this criterion has also been used in robotic applications [52]. With this criterion, a constraint is derived as

$$\tau_{rmc} \leq T_g \tag{4.23}$$

where $\tau_{rmc} = \sqrt[3]{\frac{1}{\Delta t}\int_0^{\Delta t} \tau^3(t)\mathrm{d}t}$, with $\tau(t)$ being the required torque from the gearbox output. T_g is the limit for rated torque of the gearbox.

Maximum output torque limit. The required peak torque τ_g with respect to the output side has to be smaller than or equal to the allowable peak torque T_g^{max} of the harmonic drive

$$\tau_g \leq T_g^{max} \tag{4.24}$$

where $\tau_g = \max\{|\tau(t)|\}$.

Maximum permissible input speed limit. The required maximum input peak speed n_{in} has to be smaller than or equal to the maximum

permissible input speed N_g^{max} of a gearbox

$$n_{in} \leq N_g^{max} \tag{4.25}$$

where $n_{in} = \max\{|\dot{\theta}(t) \cdot \rho|\}$.

4.3.5 Objective function formulation

Substituting Eq. (4.19) into Eqs. (4.20-4.21) and expanding Eqs. (4.22-4.25) yield the constraints on the motors and gearboxes. The objective function, $f(\mathbf{x})$, is formulated as

$$\min_{\mathbf{x}} \quad f(\mathbf{x}) = \sum_{i=1}^{5} \{m_m(\mathbf{u}_m) + m_g(\mathbf{u}_g)\}_i \tag{4.26a}$$

$$\mathbf{x} = [\mathbf{u}_m, \mathbf{u}_g, \mathbf{u}_d]$$

subject to

$$C_{min} \leq GCI(\mathbf{u}_d) \tag{4.26b}$$

$$T_{m,i} \geq \sqrt{\frac{1}{\Delta t}\int_0^{\Delta t} \left\{(J_m(\mathbf{x}) + J_g(\mathbf{x}))\ddot{\theta}(t)\rho + \frac{\tau(t,\mathbf{x})}{\rho\eta_g}\right\}_i^2 \cdot \mathrm{d}t} \tag{4.26c}$$

$$T_{m,i}^{max} \geq \max\left\{\left|(J_m(\mathbf{x}) + J_g(\mathbf{x}))\ddot{\theta}(t)\rho + \frac{\tau(t,\mathbf{x})}{\rho\eta_g}\right|\right\}_i \tag{4.26d}$$

$$N_{m,i}^{max} \geq \max\left\{|2\pi\dot{\theta}(t) \cdot \rho|\right\}_i \tag{4.26e}$$

$$T_{g,i} \geq \sqrt[3]{\frac{1}{\Delta t}\int_0^{\Delta t} \tau_i^3(t,\mathbf{x}) \cdot \mathrm{d}t} \tag{4.26f}$$

$$T_{g,i}^{max} \geq \max\{|\tau(t,\mathbf{x})|\}_i \tag{4.26g}$$

$$N_{g,i}^{max} \geq \max\left\{|\dot{\theta}(t) \cdot \rho|\right\}_i \tag{4.26h}$$

where $T_{m,i}$, $T_{m,i}^{max}$, and $N_{m,i}^{max}$ are the nominal torque, stall torque, and maximum speed of the motor in joint i. Moreover, $T_{g,i}$, $T_{g,i}^{max}$, and $N_{g,i}^{max}$ are the rated output torque, maximum output torque, and maximum input speed of the gearbox in joint i. Among these constraints, Eqs. (4.26c-4.26e) apply to the motor selection, while Eqs. (4.26f-4.26h) are for the gearbox selection. The kinematic performance is constrained through Eq. (4.26b). So far, we have formulated the design problem as a discrete optimization problem, which can be solved by commercial available codes. We select a non-gradient method called Complex for this purpose. The implementation is outlined in the next section.

4.4 Procedure of optimization

After the problem of optimization is formulated, a discrete optimization algorithm, the Complex method, is used to solve the problem.

4.4.1 Optimization by the Complex method

The Complex method is a non-gradient based optimization method, first presented by Box [31]. With this method, a number of points (sets of design variables) will be evaluated against the objective function. The set of design variables minimizing the objective function is denoted as the best point $\mathbf{x}_b$, while the one maximizing the objective function is denoted as the worst point $\mathbf{x}_w$. Their corresponding values of the objective function are noted as the best and worst values. The candidate point is found by the reflection of the worst point through the centroid $\mathbf{x}_c$ with a reflection coefficient α (as shown in Fig. 4.4), yielding the following expression for the candidate design point.

$$\mathbf{x}_c = \frac{1}{m-1}\sum_{i=1}^{m}\mathbf{x}_i, \quad \mathbf{x}_i \neq \mathbf{x}_j \tag{4.27a}$$

$$\mathbf{x}_{cand}^{old} = \mathbf{x}_c + \alpha(\mathbf{x}_c - \mathbf{x}_w) \tag{4.27b}$$

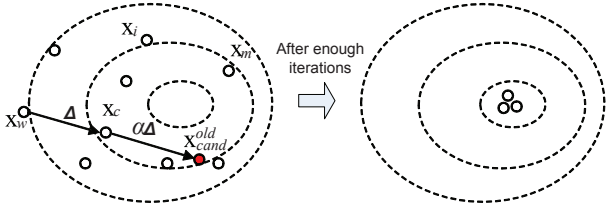


Figure 4.4: Illustration of the Complex method.

To avoid converging at a local minimum, the candidate point can be modified as

$$\begin{aligned}\mathbf{x}_{cand} = \frac{1}{2}\left(\mathbf{x}_{cand}^{old} + \varepsilon\mathbf{x}_c + (1-\varepsilon)\mathbf{x}_b\right) \\ + (\mathbf{x}_c - \mathbf{x}_b)(1-\varepsilon)(2K-1)\end{aligned} \tag{4.28}$$

where K is a random number varying in the interval $[0, 1]$, with

$$\varepsilon = \left(\frac{n_r}{n_r + k_r - 1}\right)^{\frac{n_r+k_r-1}{n_r}}$$

Here k_r is the number of repeating times the point has repeated itself, and n_r is a parameter which is recommended as 4 in the program. The algorithm converges when the difference between the best and worst objective function values is less than a user defined tolerance.

4.4.2 Design variable programming

The design points in the Complex method are usually not integers. On the other hand, the design variables $\mathbf{u}_m$ and $\mathbf{u}_g$ have to be integral, since they are the index numbers from the categories of motors and gearboxes. Hereby, a round function is introduced to transfer the design variables into integral numbers. The round function is given as

$$\begin{aligned} x_{DV} &= round(x) \\ &= \begin{cases} x_{int}; & \text{if } x_{int} \le x < x_{int} + 0.5 \\ x_{int} + 1; & \text{if } x_{int} + 0.5 \le x < x_{int} + 1 \end{cases} \end{aligned} \tag{4.29}$$

where x is a design variable manipulated by the Complex method, x_{int} is the integral part of the number x, and x_{DV} is the rounded design variable. x_{DV} is used to update link lengths and the mass of motors and gearboxes in inverse kinematic and dynamic analysis.

4.4.3 The optimization routine

The implementation of the optimization takes two steps: implementation of the optimization routine and generation of a parametric simulation model. The optimization program is implemented in Matlab. The flow diagram of the optimization routine is shown in Fig. 4.5.

4.5 The arm design optimization

4.5.1 Initial arm trajectory

Design optimizations were conducted for the robot arm, using the integrated dimensional and drive-train optimization method. The initial arm trajectory in the base coordinate system is defined as $X_{ef}(t) = 50 + 400(1 - \cos(t))$, $Y_{ef}(t) = -1000 + 800(1 - \cos(t/2))$, and $Z_{ef}(t) = 280 + 250(\cos(t/2) - 1)$, all with unit of mm. The Euler angles for the end-effector are given as $[0, \cos(t/20), 0]$, which implies the end-effector remains horizontal during the motion. The payload is defined as a point of mass of $5kg$.

4.5.2 Parameterized dimension

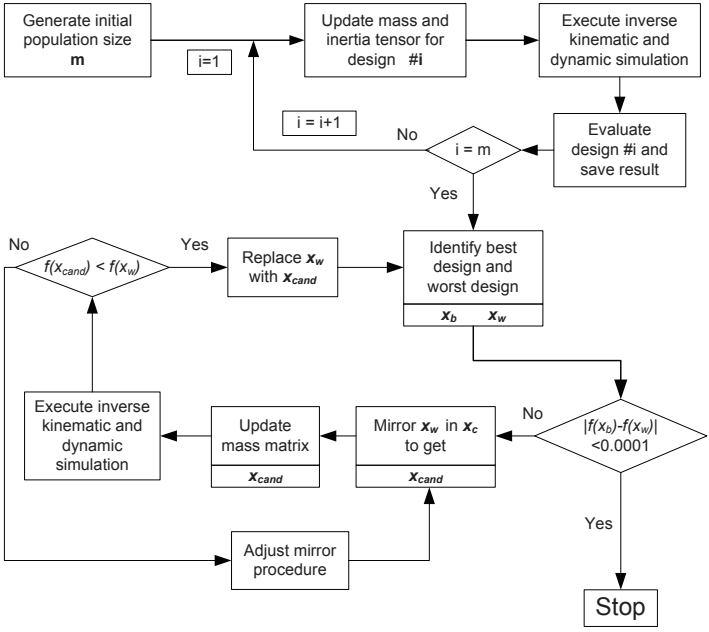


Figure 4.5: Diagram of the optimization routine.

The variable geometric dimensions include the lengths of the upper arm l_1 and lower arm l_2 (in Fig. 4.2), while h_1 and d_1 are fixed. To keep the reachable space of the robotic arm constant, the total reaching distance $L = l_1 + l_2$ is fixed.

One non-dimensional parameter r is introduced as $r = l_1/L$. Considering the structural issues, a minimum length is required for both lower and upper arms, which means

$$r \in [r_{min}, r_{max}] \tag{4.30}$$

The link length ratio r is manipulated in the interval $[r_{min}, r_{max}]$, so there is infinite number of ratios theoretically. In practice, a vector $\mathbf{r}$ is defined by discretizing r in the interval $[r_{min}, r_{max}]$ with a step of $Z = 0.05$.

$$\mathbf{r} = \{r_{min} + u_d \cdot Z\}_{u_d=1}^{u_d=c} \tag{4.31}$$

where u_d is an index number for this length ratio, and $c = (r_{max} - r_{min})/Z + 1$. In the case of optimization with multiple dimensions, u_d becomes an array of indices, $\mathbf{u}_d$, for dimensional variables.

4.5.3 Candidate components

The candidate components, including motors and gearboxes, can be defined by the designer on the basis of available products. In this work, the components are to be selected from Maxon motors and Harmonic drives. To this end, nine candidate motors are selected. They are listed in the database ascendingly with respect to the mass of motor, as shown in Table 4.2. Moreover, four gearboxes from Harmonic Drive CPU units are selected and included in the database, as listed in Table 4.3. For the Harmonic Drive gearboxes, the efficiency is a function of operation speed. In this paper, the gear efficiency is set to 0.85 for all gearboxes, which is an average value from product catalog.

Table 4.2: Candidate motor data from Maxon Motor [53].

Index No.	Motor Model	T_m [Nm]	T_m^{max} [Nm]	N_m^{max} [rpm]	J_m $[g \cdot cm^2]$	m_m [kg]
1	RE 25	0.0284	0.28	14000	10.5	0.13
2	RE 26	0.0321	0.227	14000	12.1	0.15
3	EC-i 40	0.0667	1.81	15000	24.2	0.21
4	RE 30	0.0882	1.02	12000	34.5	0.238
5	EC 32	0.0426	0.353	25000	20	0.27
6	RE 35	0.0965	0.967	12000	67.4	0.34
7	RE 36	0.0795	0.785	12000	67.2	0.35
8	EC 40	0.127	0.94	18000	85	0.39
9	RE 40	0.184	2.5	12000	138	0.48

Table 4.3: Candidate gearbox data from Harmonic Drive [54].

Index No.	CPU Unit Size	Ratio	T_g [Nm]	T_g^{max} [Nm]	N_g^{max} [rpm]	J_g $[kg \cdot m^2]$	m_g [kg]
1	14	100	11	54	8500	0.033×10^{-4}	0.54
2	17	100	39	110	7300	0.079×10^{-4}	0.79
3	20	100	49	147	6500	0.193×10^{-4}	1.3
4	25	100	108	284	5600	0.413×10^{-4}	1.95

The gear ratio of each joint is set to $\rho = \{200, 200, 200, 51, 100\}$, orderly from Joint 1 to 5. Note there are two stage gearboxes in Joints 1, 2 and 3, a planetary gearhead and a Harmonic Drive unit. For simplicity, only the mass of the Harmonic Drive gearbox is parameterized, while the mass of the planetary gearhead is set to constant. The Harmonic Drive CPU unit

is used in all joints except Joint 4, due to the joint structure consideration. A planetary gearhead is used in Joint 4, so $u_{g,4} = 0$.

4.5.4 Optimization results

Once the candidate components have been selected, their corresponding limits of the inequalities (4.26c-4.26h) are determined. The limit of kinematic performance GCI is set to $C_{min} = 0.02$, a limit that can be satisfied by a robotic arm with link ratios between $r = 0.2 \sim 0.8$. Optimized designs of motors and gearboxes for the robotic arm are listed in Table 4.4. The initial combination of motors and gearboxes are selected based on previous dynamics simulation of the robotic arm. The optimized mass of the robotic arm is 9.92 kg, with a reduction of 41% corresponding to the initial combinations. Another optimization case with fixed link length $r = 0.5$ is also shown in Table 4.4 for comparison. In this design case, the mass change is not significant relative to the previous optimized case, but still noticeable. The change is due to the size of the motor at Joints 1 and 3. Referring to Table 4.2, the corresponding nominal torques are 0.0426 Nm (EC 32) and 0.0965 Nm (RE 35) for Joint 1.

Table 4.4: Optimal link ratio and drive-train combinations for minimization of arm mass.

Joint	Initial		Optimized		Fixed $r = 0.5$	
	Motor	Gearbox	Motor	Gearbox	Motor	Gearbox
1	RE 40	CPU 17	EC 32	CPU 14	RE 35	CPU 14
2	RE 35	CPU 17	RE 25	CPU 14	RE 25	CPU 14
3	RE 35	CPU 17	RE 30	CPU 14	RE 35	CPU 14
4	RE 35	Gearhead	RE 25	Gearhead	RE 25	Gearhead
5	RE 35	CPU 17	RE 25	CPU 14	RE 25	CPU 14
Link ratio	$r = 0.5$		$r = 0.6$		$r = 0.5$	
Arm mass[kg]	16.7		9.92		9.98	

The convergence of the objective function is depicted in Fig. 4.6, both best (black dot) and worst (gray dot) values from the Complex algorithm are shown. The solution to the optimal result is achieved at 3500 iterations with 130 population sizes. In this work, the tolerance of convergence is 0.0001.

The convergences of the link length ratio and GCI are shown in Figs. 4.7 and

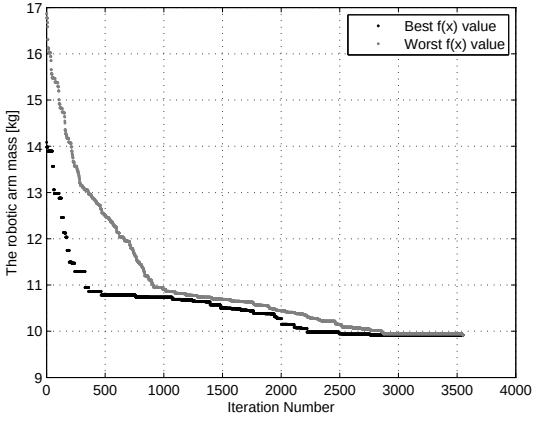


Figure 4.6: Convergence of the mass of the robotic arm.

4.8. The link length ratio is converged to $r = 0.6$. Figure 4.9(a) illustrates the convergence of motor design variables. Only the convergence plots for Joints 1 and 5 are displayed for clarity. The convergence of gearbox design variables is depicted in Fig. 4.9(b). Comparing the convergence rates for the motor and gearbox design variables, the gearbox design variables converge faster towards the optimal results than the motor design variables. This phenomena is caused by that the mass difference among Harmonic Drive units is larger than among motors.

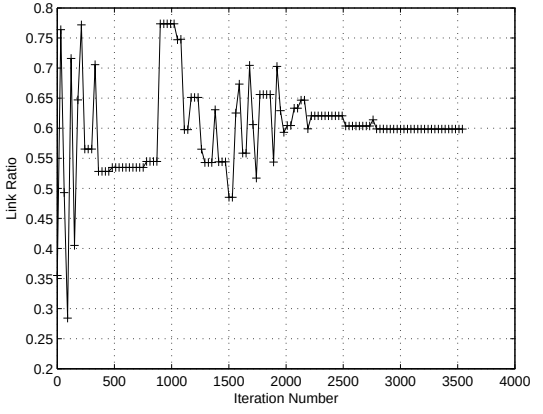


Figure 4.7: Convergence of the link length ratio.

The variations of motor torques of Joints 1 and 2 for the initial and the optimal design are shown together in Fig. 4.10. The torques of the optimal

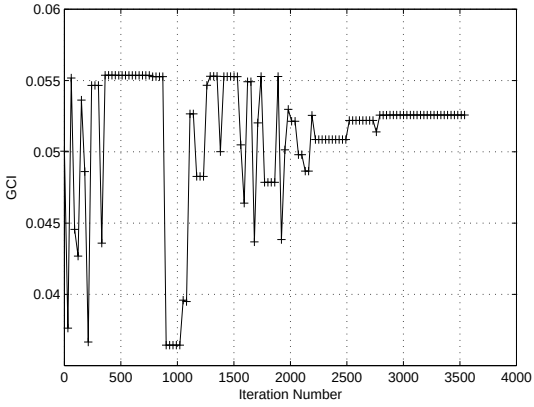
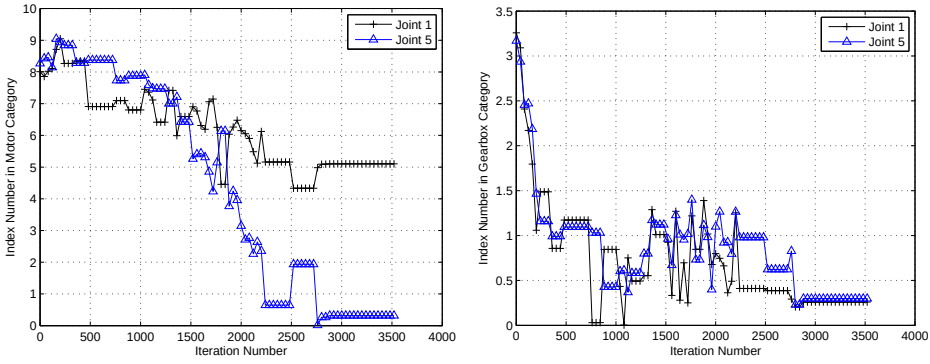


Figure 4.8: Convergence of the GCI of the arm.



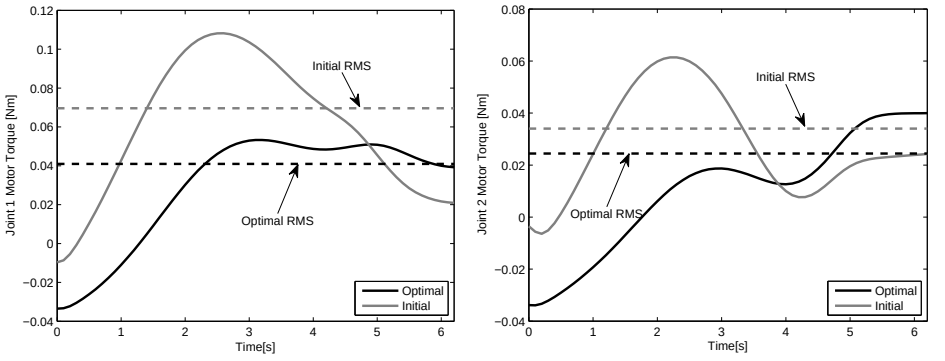
(a) Motors (b) Gearboxes

Figure 4.9: Convergence plots for the design variables of motors and gearboxes.

design are depicted in black color, and that of the initial design are in gray. The RMS value of each torque is depicted with dashed line. It is seen that the optimal design has a reduction of 41.29% RMS torque for Joint 1, and a reduction of 26.87% RMS torque for Joint 2.

4.5.5 Design optimization with an alternative trajectory

Another trajectory is used for the integrated optimization. This trajectory is given for a pick-and-place operation (PPO) defined by $X_{ef}(t) = 600$, $Y_{ef}(t) = -150\cos(t) - 150$, and $Z_{ef}(t) = 300\cos(t/2) - 100$, all with unit of mm, as depicted in Fig. 4.11. The duration is 6.2 seconds. The orientation for the end-effector is described by Euler angles $[0, \cos(t/20), 0]$, following



(a) Joint 1 (b) Joint 2

Figure 4.10: Motor torques for initial and optimal drive-train combinations.

the Z-X-Z convention.

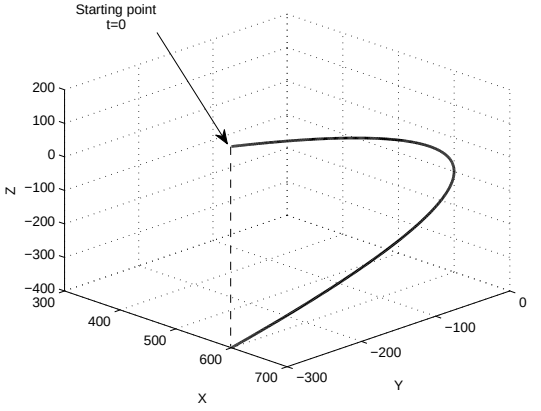


Figure 4.11: Illustration of the alternative trajectory.

Convergence of the mass of the robotic arm is depicted in Fig. 4.12. The solution to the optimal result is achieved at 3800 iterations with 130 population sizes. The optimization result for the alternative trajectory is listed in Table 4.5. The optimization with the alternative trajectory yields a design of mass slightly less than the case with the initial trajectory.

4.5.6 Optimization with a different GCI limit

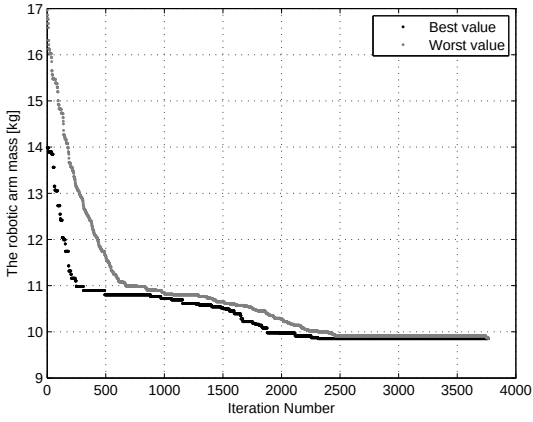


Figure 4.12: Convergence of the mass of the robotic arm for the alternative trajectory.

Table 4.5: Optimization results for an alternative trajectory.

Joint	Motor	Gearbox
1	EC 32	CPU 14
2	EC-i 40	CPU 14
3	RE 25	CPU 14
4	RE 25	Gearhead
5	RE 25	CPU 14
Link ratio	$r = 0.6$	
Arm mass[kg]	9.85	

Another optimization is conducted with $C_{min} = 0.05$. The trajectory utilizes the same one in Sec. (4.5.4). The convergence of the objective function is depicted in Fig. 4.13. The optimized mass is 9.88 kg. The solution to the optimal result is achieved at 3500 iterations, which implies the same converging rate as the case with $C_{min} = 0.02$.

The link length ratio is converged to $r = 0.6$. The variance of GCI during the optimization is shown in Fig. 4.14. The convergence result with $C_{min} = 0.05$ is identical to the one with $C_{min} = 0.02$, as shown in Figs. 4.6 and 4.8.

4.5.7 Discussions

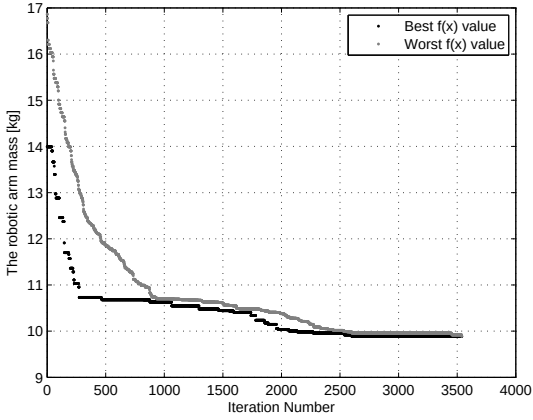


Figure 4.13: Convergence of the mass of the robotic arm for $C_{min} = 0.05$.

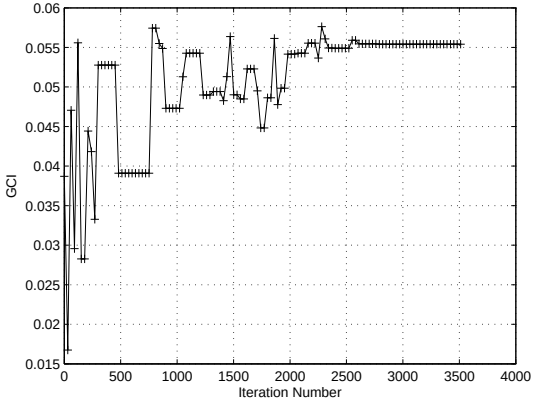


Figure 4.14: Convergence of the GCI of the arm for $C_{min} = 0.05$.

By comparing the optimization results for two trajectories (Tables 4.4 and 4.5), it can be seen that the minimum mass changes slightly, while the integrated optimization method yields different combinations of motors and gearboxes. On the other hand, the link length ratio of the arm structure remains unchanged. This suggests that the design optimization with the selected trajectories is practical. The results generated from a worst case identified by the first trajectory is able to fulfill dynamic requirements in normal manipulations.

The optimal link length ratio for both trajectories and different GCI limits is $r = 0.6$. Comparing this ratio to some robotic manipulators [61]

such as KUKA-KR-R650 ($r = 0.55$), Denso-VM-6083D series ($r = 0.54$), Mitsubishi-RV-2AJ series ($r = 0.61$) and Staubli-TX40 series ($r = 0.58$), the optimal ratio agrees generally with these industrial robots. The difference between these ratios can be considered as the influence on arm shape and mass distributions. The prototype of the 5-dof robotic arm in this work is shown in Fig. 4.15. The components of drive-train in the prototype are selected and scaled based on the optimization results.

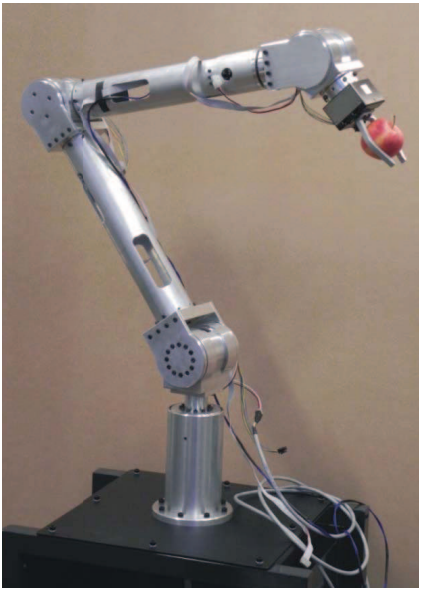


Figure 4.15: Prototype of the 5-dof robotic arm.

While the integrated dimensional and drive-train design optimization was done on a platform developed with Matlab, the method can be applied to commercial available CAD/CAE systems. To this end, dynamics simulation will be implemented in a CAE system, e.g. MSC.ADAMS™. On the other hand, Matlab programme will serve as an interface between the user and CAD/CAE systems and run the Complex routine. In this way, the method can easily be extended to include other constraints, for example, strength constraints evaluated by FEA software like ANSYS™.

The proposed approach is aimed for the off-the-shelf design of a robotic arm, for which the drive-train components are standard commercial products. This implies that the performance (weight) improvement with the developed method may be limited for light-weight robot designs with customer

designed actuators, as the case of DLR light-weight arm, for which the challenges lie in the novel motor design, topology optimization and new materials, rather the problems addressed in this work.

4.6 Conclusions

An integrated dimensional and drive-train optimization method was proposed for the design of robotic manipulators. Selections of geometric dimensions, motors and gearboxes were formulated as a discrete optimization problem, which was solved by a non-gradient optimization method. Global conditioning index was taken as constraint on kinematic performance of the robot. The robot dynamics was constrained by considering characteristics of motors and gearboxes. The proposed method is able to reach a design with lower mass and optimal geometric dimensions. A 5-dof light-weight anthropomorphic robotic arm was designed by implementing the presented method. Case studies were conducted to demonstrate the application of the method in the design of robotic manipulators. The optimal design is able to fulfill dynamic requirements in normal operations. In the future works, constraints will be extended to include considerations such as strength/stiffness and energy consumptions.

5

Article III

An Integrated Approach to the Design and Optimization of Lightweight Robotic Arms

The paper has been submitted as:

Lelai Zhou, Shaoping Bai, and Michael R. Hansen, "An Integrated Approach to the Design and Optimization of Lightweight Robotic Arms", *IEEE Transactions on Robotics*, (submitted in July, 2011)

Abstract

This paper proposed a new approach to the design and optimization of lightweight robotic arms, where robot kinematics, dynamics, drive-train design and strength analysis by means of finite element analysis (FEA) are considered. The approach makes use of a discrete parameterized robotic model, in which kinematic dimensions, structural dimensions, the motors and gearboxes from commercially available components are parameterized as design variables for design analysis and optimization. Constraints are formulated on the basis of kinematic performance, dynamic requirements and structural strength constraints, whereas the main objective is to minimize the weight. The Complex method is selected for the discrete optimization problem. An integrated design platform is developed to implement the proposed approach, which enables robot design analysis and optimization. The proposed approach is demonstrated with a design example of a five degree-of-freedom lightweight arm for assistive applications.

5.1 Introduction

Light and strong robotic arms are desirable for many applications where robots interact with human closely. Examples of light-weight design can be found in a number of robots such as the DLR robotic arm [4], WAM manipulator by Barrett Technology [8], among others. However, the design of lightweight robots faces many challenges, varying from power supply, actuators, power transmission and structural parts. New methods are required to take these challenges for lightweight designs.

Among factors that can contribute to a lightweight design, the drive trains and the structural parts are considered as the main two influencing the mass reduction. A majority of research work in design optimization are related to the drive-train design. An early attempt on drive-train design optimization can be found in [18], in which Chedmail and Gautier proposed a method for the optimum selection of robot actuators to minimize the total mass of all the actuators. Pettersson and Ölvander [19] reported a method of design optimization, in which the drive-train of two joints were optimized for an industrial manipulator. A simulation environment called Modelica with robot optimization facilities was presented, where the parameters of a controller were tuned by a multi-criteria parameter optimization method to improve the system dynamics [20]. A drive-train design optimization method was reported in [63], which is able to

optimally select combinations of motors and gearboxes from a catalogue of commercially available components for each dof of a robot arm.

Lightweight design of structural parts can contribute a large reduction to the weight of the system. Regarding structural components optimization, finite element analysis (FEA) is widely used. FEA was utilized to conduct structural optimization in the design of humanoid robots [28]. The utilization of FEA in robotic arm design and structural optimization can be found in [29, 30].

On the other hand, dimensional optimization was studied for improvement of robotic performance, either kinematic or dynamic one. An integrated structure-control design optimization method of a two-link flexible robot arm was presented, where the structural and control parameters were optimized simultaneously [23]. An optimal design of manipulator parameter using evolutionary optimization method was proposed in [24]. Optimal dynamic performance based methods was reported in [26], among others. It can be noticed that structural dimensions of robotic manipulators were rarely considered. Moreover, dimensional and drive-train optimizations were mostly conducted separately. An integrated approach is desired in order to fully utilize the potential of applying optimization techniques to robot design.

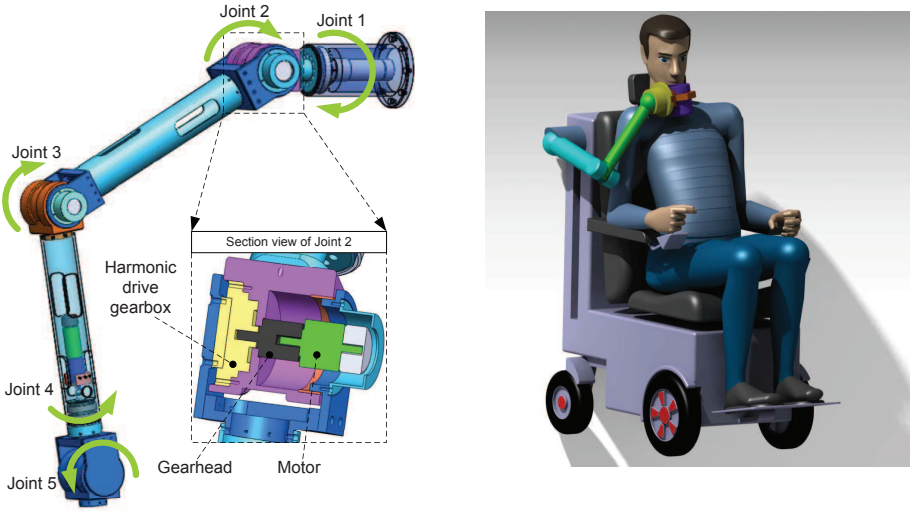
In this work, an integrated design and optimization method is developed for the design of lightweight robotic manipulators. The method combines the kinematics, dynamics and structural strength analysis in a single design stage, while the main objective is to minimize the weight of the robot. The proposed method extends the integrated design optimization method reported in authors' previous work [64]. In the new method, the structural dimensions of a robotic arm are taken as variables in the design optimization, in addition to the parameters of the drive-trains. The arm structure and the drive-train will be optimized to obtain light-weight robotic arm, while constraints on the kinematics, drive-train dynamics and structural strength are considered. The paper shows that the integrated optimization method can contribute to further reducing the arm weight.

The paper is organized as follows: Section 5.2 introduces a robotic arm and the parameterized model. Kinematic, drive-train and structural strength constraints are developed in Sections 5.3, 5.4 and 5.5, respectively. Section 5.6 presents the formulation of the integrated design optimization. Section 5.7 illustrates the optimization procedure of the proposed method. A design

example is given in Section 5.8, followed by conclusions in Section 5.9.

5.2 A robotic arm

The light-weight robotic arm considered in this paper has five degrees of freedom (dof), with two dof at the shoulder, one at the elbow, and two at the wrist, as shown in Fig. 5.1(a). This is a human-like arm design, which is to be mounted on an electric wheelchair to assist disabled in simple manipulations like picking, placing, door opening, etc. For this purpose, a gripper is employed at the end of the arm, as demonstrated in Fig. 5.1(b).



(a) (b)

Figure 5.1: A 5-dof light-weight anthropomorphic arm: (a) CAD model, (b) a scenario of the intended application.

5.2.1 Arm mechanism

A modular approach is adopted in the design. CPU series gearboxes of Harmonic Drive™ are used as transmission elements and, simultaneously, as the mechanical joints, for the different dof. To increase the torque capabilities of Joints 1, 2 and 3, a second stage of gearhead is used between Harmonic Drive and the motor. The geared motors and Harmonic Drive gearboxes are mounted inside the joint housings, while the axes of rotation

coincide with the joint axes. The physical realization of Joint 2 is illustrated in Fig. 5.1(a).

5.2.2 Parameterized dimensions

The structural parts of the robotic arm are to be optimized in the integrated optimization method. Some structural dimensions are parameterized, where l_1 and l_2 are lengths of the upper and lower arms.

Figure 5.2 shows some parameterized dimensions of the robotic arm. These dimensions fall into two groups: the assembling dimensions including the link lengths of the upper arm l_1 and the lower arm l_2, and the structural dimensions displayed in Fig. 5.2(b). The assembling dimensions determine the robotic arm's kinematic performance, while the structural dimensions affect the arm structural strength. The descriptions of the parameterized dimensions are listed in Table 5.1.

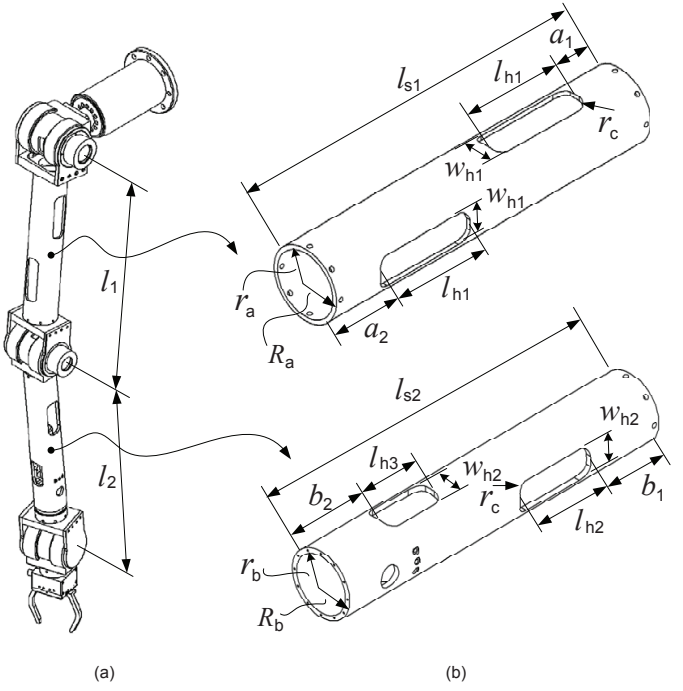


Figure 5.2: Dimensional parameters of the robotic arm.

To keep the reachable space of the robotic arm constant, the total reaching distance $L = l_1 + l_2$ is fixed. One non-dimensional parameter r is introduced

Table 5.1: Structural parameters of the robotic arm [mm].

Upper arm	Forearm	Parameter descriptions
l_{s1}	l_{s2}	tube length
r_a	r_b	inner radius
R_a	R_b	outer radius
w_{h1}	w_{h2}	widths of the opening slots
l_{h1}	l_{h2}, l_{h3}	lengths of the opening slots
a_1, a_2	b_1, b_2	lengths used to position the slots

as $r = l_1/L$. Considering the structural issues, a minimum length is required for both lower and upper arms, which means $r \in [r_{min}, r_{max}]$.

The lengths of arm links are subject to the size of joints. In this design, we take $l_{s1} = l_1 - 150$ [mm] and $l_{s2} = l_2 - 150$ [mm]. Other dependent dimensions are determined by: $l_{h1} = (l_{s1} - a_1 - a_2 - 30)/2$ [mm], $l_{h3} = l_{h2}/2$ [mm], $l_{h2} = 2(l_{s2} - b_1 - b_2 - 30)/3$ [mm].

The dimensional design variables include r, r_a, r_b, w_{h1}, and w_{h2}, as shown in Fig. 5.2(b). In practice, an array of dimension $\mathbf{u}$ is defined by discretizing r, r_a, r_b, w_{h1} and w_{h2} from each interval with a step e.

$$\mathbf{u} = \{u_{min} + u_d \cdot e\}_{u_d=1}^{u_d=c} \tag{5.1}$$

where $\mathbf{u}_d = [u_r, u_{r_a}, u_{r_b}, u_{w_{h1}}, u_{w_{h2}}]$, and $c = (u_{max} - u_{min})/e + 1$.

5.2.3 Jacobian matrix

The joint angular velocity can be calculated with the Jacobian matrix

$$\dot{\boldsymbol{\theta}} = \mathbf{J}^{-1}\mathbf{v}_{ef} \tag{5.2}$$

where $\dot{\boldsymbol{\theta}} = [\dot{\theta}_1, \dot{\theta}_2, \ldots, \dot{\theta}_n]^T$ is an n-dimensional (n denotes the number of dof) vector of the joint angular velocities, $\mathbf{J}$ the Jacobian matrix, and $\mathbf{v}_{ef}$ the velocity of the end-effector.

For a revolute joint, the Jacobian matrix can be calculated by [39]

$$\mathbf{J} = [\mathbf{J}_1, \mathbf{J}_2, \ldots, \mathbf{J}_n] \tag{5.3}$$

with

$$\mathbf{J}_i = \begin{bmatrix} \mathbf{z}_{i-1} \times \mathbf{p}_{i-1} \\ \mathbf{z}_{i-1} \end{bmatrix}, \quad i = 1, \ldots, n \tag{5.4}$$

where $\mathbf{z}_{i-1}$ and $\mathbf{p}_{i-1}$ are given by:

$$\mathbf{z}_{i-1} = \mathbf{R}_{i-1}\,[0 \;\; 0 \;\; 1]^T \tag{5.5a}$$

$$\mathbf{p}_{i-1} = \mathbf{R}_{i-1}\mathbf{q}_{i-1} + \mathbf{p}_i \tag{5.5b}$$

with $\mathbf{q}_{i-1} = [a_i \cos\theta_i, a_i \sin\theta_i, d_i]^T$. Matrix $\mathbf{R}_{i-1}$ denotes the rotation matrix from the reference coordinate system to the $(i-1)$th coordinate system. The parameters a_i and d_i are defined following the D-H convention, i.e., a_i is offset distance between two adjacent joint axes, and d_i is translational distance between two incident normals of a joint axis. When the desired end-effector velocity $\mathbf{v}_{ef}$ is given, the joint angular velocity can be solved by Eq. (5.2).

5.2.4 Inverse dynamics

The integrated structural and drive-train optimization utilizes a dynamic model of the robotic arm for dynamic evaluations. The governing equation of the arm motion can be written as:

$$\mathbf{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \mathbf{v}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \mathbf{g}(\boldsymbol{\theta}) = \boldsymbol{\tau} \tag{5.6}$$

where $\mathbf{M}$ is the mass matrix, $\mathbf{v}$ is the vector of Coriolis and centrifugal terms of the links, $\mathbf{g}$ is the vector of gravitational forces, and $\boldsymbol{\tau}$ is the vector of joint torques. The mass matrix $\mathbf{M}$ of Eq. (5.6) is not constant. Instead, it changes with arm poses, arm dimensions and mass distributions.

5.3 Kinematic constraints

The integrated optimization is proposed to minimize the weight of the robotic arm with constraints on kinematics performance, drive-train, and the structural strength. The selection of a drive-train is constrained through the dynamic equation and the selecting criteria for motors and gearboxes. The structural dimensions influence the robotic dynamics. On the other hand, they also determine the kinematic performance of the robotic manipulator. This may be formulated as a constraint on the kinematic performance index as described presently.

The kinematics performance is one of the major concerns in robot design. It is desirable for a robot to have a high kinematics performance, while the drive-drain is optimized. Several performance indices are available for the design of robotic manipulators. They include manipulability measure

proposed by Yoshikawa [56] and the global conditioning index (GCI) by Gosselin and Angeles [57]. The GCI is considered in this work.

The GCI within a workspace W is defined as

$$GCI = \frac{\int_W \frac{1}{\kappa} \mathrm{d}W}{\int_W \mathrm{d}W} \tag{5.7}$$

with the condition number κ given by

$$\kappa = \| \mathbf{J}(\boldsymbol{\theta}, \mathbf{u}_d) \| \| \mathbf{J}^{-1}(\boldsymbol{\theta}, \mathbf{u}_d) \| \tag{5.8}$$

where $\mathbf{J}(\boldsymbol{\theta}, \mathbf{u}_d)$ is the Jacobian matrix defined in Eq. (5.2), $\boldsymbol{\theta}$ is the vector of joint angles, and $\mathbf{u}_d$ is an array of structural dimensions. The Euclidean norm $\| \cdot \|$ of the matrix is defined as

$$\| \mathbf{J} \| = \sqrt{tr(\mathbf{J}^T \mathbf{N} \mathbf{J})} \tag{5.9}$$

with $\mathbf{N} = \frac{1}{n}\mathbf{I}$, where n is the dimension of the square matrix $\mathbf{J}$, and $\mathbf{I}$ is the $n \times n$ identity matrix. An example showing the relation between the inverse of condition number $1/\kappa$ and the joint angles θ_2 and θ_3 is depicted in Fig. 5.3.

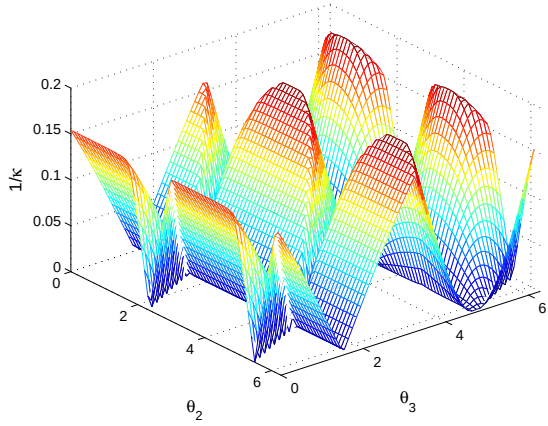


Figure 5.3: The inverse of condition number $1/\kappa$ with respect to θ_2 and θ_3.

In practice, the GCI of a robotic manipulator is calculated through a discrete approach as [62]

$$GCI = \frac{1}{V} \sum_{i=1}^{m} \frac{1}{\kappa_i} \Delta V_i \tag{5.10}$$

where V is the workspace volume, and m is the number of discrete points. In the case of equal-volumetric discretization, $\Delta V_i \equiv \Delta V$, Eq. (5.10) is transformed to

$$GCI = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{\kappa_i} \tag{5.11}$$

The GCI is dimension-dependent, which means

$$GCI = GCI(\mathbf{u}_d) \tag{5.12}$$

To keep a high kinematics performance with selected link lengths in the integrated optimization, a constraint is given on the GCI

$$GCI(\mathbf{u}_d) \geq C_{min} \tag{5.13}$$

where C_{min} is the minimum acceptable GCI.

5.4 Drive-train constraints

A drive-train model of a single joint is shown in Fig. 5.4. The drive-train consists of a motor, a linkage and a gearbox for speed reduction. Taking into account of gear efficiency, the required motor torque for the ith joint can be calculated by

$$\tau_{m,i} = \left\{ (J_m + J_g)\ddot{\theta}(t)\rho + \frac{\tau(t)}{\rho\eta_g} \right\}_i ; \quad i = 1, \ldots, 5 \tag{5.14}$$

where ρ_i is the gear ratio. $J_{m,i}$ is mass moment of inertia of the ith motor; $J_{g,i}$ is the equivalent mass moment of inertia of the ith gearbox; $\eta_{g,i}$ is the corresponding gear efficiency, and $\tau_i(t)$ is the load at the output link which can be solved by (5.6).

5.4.1 Motor selection criteria

Motors for robotic arms are usually selected from two motor groups, brushed and brushless DC motors. In selecting motors, the following three criteria are considered:

$$\tau_{rms} \leq T_m;\ \tau_p \leq T_m^{max};\ n_p \leq N_m^{max} \tag{5.15}$$

where τ_{rms} denotes the root mean square (RMS) value of the required motor torque, and T_m is the nominal torque of the motor. $\tau_p = \max\{|\tau_m|\}$ is the required peak torque, and T_m^{max} is the stall torque of the motor. $n_p = \max\{|2\pi\dot{\theta}(t) \cdot \rho|\}$ is the required peak speed corresponding to the motor, and N_m^{max} denotes the maximum permissible speed of the motor.

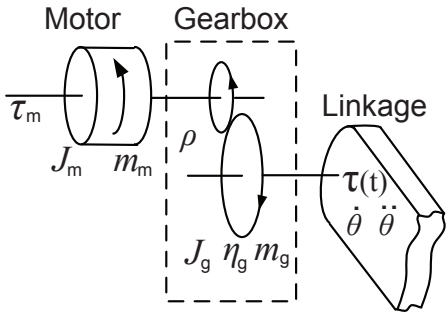


Figure 5.4: Schematic view of drive-train model for a single joint.

5.4.2 Gearbox selection criteria

For the selection of gearboxes, the following three criteria apply:

The first criterion is the RMC value of torques (τ_{rmc}), recommended by the Harmonic Drive gearbox manufacturer [50]. The RMC value is a measure of the accumulated fatigue on a structural component and reflects typical endurance curves of steel and aluminium [51]. It is therefore relevant to gearbox lifetime, and this criterion has also been used in robotic applications [52]. With this criterion, a constraint is derived as

$$\tau_{rmc} \leq T_g \tag{5.16a}$$

where $\tau_{rmc} = \sqrt[3]{\frac{1}{\Delta t}\int_0^{\Delta t} \tau^3(t)dt}$, with $\tau(t)$ being the required torque from the gearbox output. T_g is the limit for rated torque of the gearbox.

Other criteria for gearbox selection include

$$\tau_g \leq T_g^{max};\ n_g \leq N_g^{max} \tag{5.16b}$$

where $\tau_g = \max\{|\tau(t)|\}$ denotes the required peak torque with respect to the output side, and T_g^{max} is the allowable peak torque of the gearbox. $n_g = \max\{|\dot{\theta}(t) \cdot \rho|\}$ is the required maximum input peak speed, and N_g^{max} denotes the maximum permissible input speed of a gearbox.

5.5 Structural strength constraints

While the robotic arm becomes lighter, it has to meet strength requirements as well, which need to be included as constraints in optimization. In this

work, the stress and deformation of the robotic arm are considered. To maintain the strength and stiffness of the structure, the maximum von-Mises stress has to be smaller than the yield strength of the material.

$$S_{max} < S_y \tag{5.17}$$

The maximum deformation of the end-effector has to be under a relevant limit for operational consideration of the robot.

$$D_{max} < D_{lim} \tag{5.18}$$

To integrate the FEA into the optimization method, a FEA model of the robotic arm is to be built and simulated in ANSYS Workbench^{TM}. The joint structures are imported into ANSYS Workbench from CAD geometry file. The upper and lower arm links are built as parameterized model in Workbench.

The boundary conditions and loads of the robotic arm remain unchanged through the optimization iterations, as depicted in Fig. 5.5(a), where the shoulder joint of the arm is grounded. External force (payload) is applied on the gripper. A FEA simulation on the robotic arm is shown in Fig. 5.5(b).

5.6 Integrated design optimization

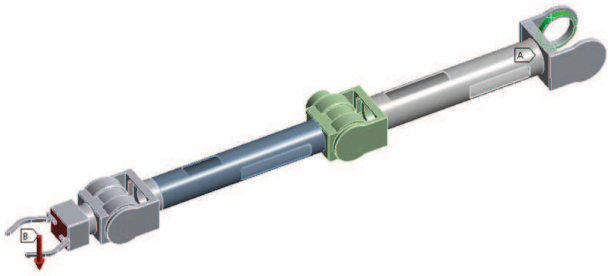
The objective of the integrated design optimization is to design a light-weight robotic arm. The task is to find the lightest combination of motor and gearbox for all five joints and the optimal link lengths that fulfill all constraints associated with the kinematic, strength and drive-train constraints. The optimization will also minimize the mass of the robotic structure (m_{arm}) by selecting optimal dimensions that fulfill constraints on the structural statics. The objective function, $f(\mathbf{x})$, is defined as

$$\begin{aligned} \min_{\mathbf{x}} \; f(\mathbf{x}) &= \sum_{i=1}^{n} \{m_m(\mathbf{u}_m) + m_g(\mathbf{u}_g)\}_i + m_{arm}(\mathbf{u}_d) \\ \mathbf{x} &= [\mathbf{u}_m, \mathbf{u}_g, \mathbf{u}_d] \end{aligned} \tag{5.19}$$

S.T.

Kinematic constraint:

$$C_{min} \leq GCI(\mathbf{u}_d) \tag{5.20a}$$



(a)

(b)

Figure 5.5: (a) Boundary conditions of the FEA model, (b) FE analysis on the robotic arm.

Strength constraints:

$$S_y > S_{max}(\mathbf{u}_d) \tag{5.20b}$$

$$D_{lim} > D_{max}(\mathbf{u}_d) \tag{5.20c}$$

Drive-train constraints:

$$T_{m,i} \geq \sqrt{\frac{1}{\Delta t}\int_0^{\Delta t}\left\{J_m(\mathbf{x}) + J_g(\mathbf{x}))\ddot{\theta}(t)\rho + \frac{\tau(t,\mathbf{x})}{\rho\eta_g}\right\}_i^2 \cdot \mathrm{d}t} \tag{5.20d}$$

$$T_{m,i}^{max} \geq \max\left\{\left|J_m(\mathbf{x}) + J_g(\mathbf{x}))\ddot{\theta}(t)\rho + \frac{\tau(t,\mathbf{x})}{\rho\eta_g}\right|\right\}_i \tag{5.20e}$$

$$N_{m,i}^{max} \geq \max\left\{|2\pi\dot{\theta}(t)\cdot\rho|\right\}_i \tag{5.20f}$$

$$T_{g,i} \geq \sqrt[3]{\frac{1}{\Delta t}\int_0^{\Delta t}\tau_i^3(t,\mathbf{x})\cdot \mathrm{d}t} \tag{5.20g}$$

$$T_{g,i}^{max} \geq \max\left\{|\tau(t,\mathbf{x})|\right\}_i \tag{5.20h}$$

$$N_{g,i}^{max} \geq \max\left\{|\dot{\theta}(t)\cdot\rho|\right\}_i \tag{5.20i}$$

where design variables of $\mathbf{x}$ include the index numbers of motors $\mathbf{u}_m = [u_{m,1},\ldots,u_{m,n}]$ and gearboxes $\mathbf{u}_g = [u_{g,1},\ldots,u_{g,n}]$, relative to databases containing commercially available components, and an array of dimensional variables $\mathbf{u}_d$.

5.7 Procedure of optimization

The integrated design optimization problem is solved by the Complex method, a method suitable for nonlinear and discrete optimization problems. In this section, the Complex method is briefed first, followed by the procedure of optimization.

5.7.1 Optimization by the Complex method

The Complex method is a non-gradient based optimization method [31]. With this method, a number of points (sets of design variables) will be evaluated against the objective function. The set of design variables minimizing the objective function is denoted as the best point $\mathbf{x}_b$, while the one maximizing the objective function is denoted as the worst point $\mathbf{x}_w$. Their corresponding values of objective function are noted as the best and worst values. After each evaluation, a candidate point is generated by reflecting the worst point through the centroid $\mathbf{x}_c$ with a reflection coefficient α (as shown in Fig. 5.6).

$$\mathbf{x}_{cand} = \mathbf{x}_c + \alpha(\mathbf{x}_c - \mathbf{x}_w) \tag{5.21}$$

where $\mathbf{x}_c = \frac{1}{m-1}\sum_{i=1}^{m}\mathbf{x}_i,\ \mathbf{x}_i \neq \mathbf{x}_j$. The coefficient α is experimentally determined, which takes the value of 1.3. To avoid converging at a local minimal, the candidate point can be found through a modified approach

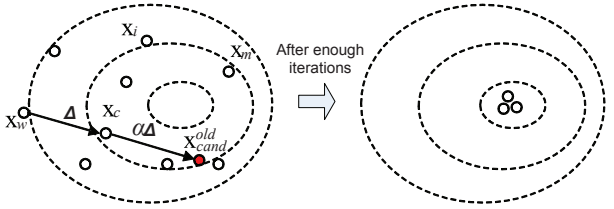


Figure 5.6: Illustration of the Complex method.

[65],

$$\begin{aligned} \mathbf{x}_{cand}^{new} &= \tfrac{1}{2}\left(\mathbf{x}_{cand}^{old} + \varepsilon\mathbf{x}_c + (1-\varepsilon)\mathbf{x}_b\right) \\ &\quad +(\mathbf{x}_c - \mathbf{x}_b)(1-\varepsilon)(2K-1) \end{aligned} \tag{5.22}$$

where K is a random number varying in the interval $[0, 1]$. Moreover,

$$\varepsilon = \beta^{-\beta}; \beta = 1 + \frac{k_r - 1}{n_r} \tag{5.23}$$

Here k_r is the number of repeating times the point has repeated itself, and n_r is a parameter which is recommended as 4 in the program. The algorithm converges when the difference between the best and worst objective function values is less than a user defined tolerance.

5.7.2 Design variables programming

The design points in the Complex method are usually continuous. However, the design variables $\mathbf{u}_m$, $\mathbf{u}_g$ and $\mathbf{u}_d$ have to be integers, since they are the index numbers from the databases of motors and gearboxes. To deal with the integer design variables, a round function is introduced to transfer the design variables into integers. The rounding function is given as

$$\begin{aligned} x_{DV} &= round(x) \\ &= \begin{cases} x_{int}; & if\ x_{int} \le x < x_{int} + 0.5 \\ x_{int} + 1; & if\ x_{int} + 0.5 \le x < x_{int} + 1 \end{cases} \end{aligned} \tag{5.24}$$

where x is the design variable manipulated by the Complex method, x_{int} is the integral part of the number x, and x_{DV} is the rounded design variable. The rounded variable x_{DV} is used to update the mass of motors and gearboxes in inverse dynamic analysis, as well as the allowable torque and speed values used to examine constraint violations.

5.7.3 The optimization routine

The integrated optimization method is implemented as a design optimization platform containing five modules, as shown in Fig. 5.7. The five modules include the CAD module, the kinematic simulation, the dynamic simulation, the FEA module, and the optimization module. Among them, the CAD module is used to build the structural model of the robotic manipulators. The kinematics simulation module is used to conduct the kinematics analysis of the robot system. Kinematics performance such as workspace (WS), global conditioning index (GCI), etc., are investigated in this module. The dynamics simulation module is used to run the dynamics analysis of a multibody system. The FEA system module deals with the structural static and dynamic analysis using finite element method. The optimization module contains algorithms that are able to deal with highly non-linear and discrete problems for running the design optimization.

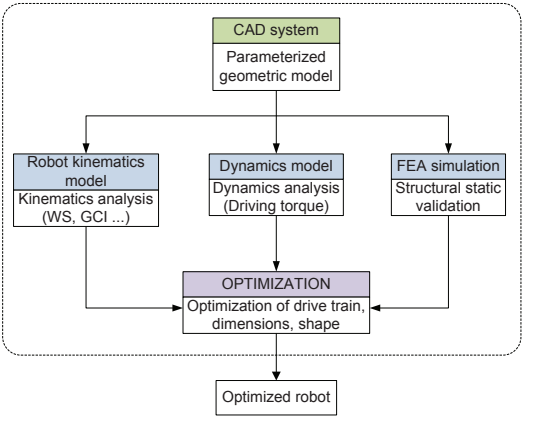


Figure 5.7: Functional modules of the integrated optimization approach.

The implementation of the integrated optimization method consists of programming of the optimization algorithm, generation of a parametric simulation model and FEA in ANSYS Workbench. Strength analysis is executed through running ANSYS Workbench by Matlab in batch mode. The kinematic and dynamic analysis, and the optimization algorithm are programmed in Matlab. The flow diagram of the optimization routine is shown in Fig. 5.8.

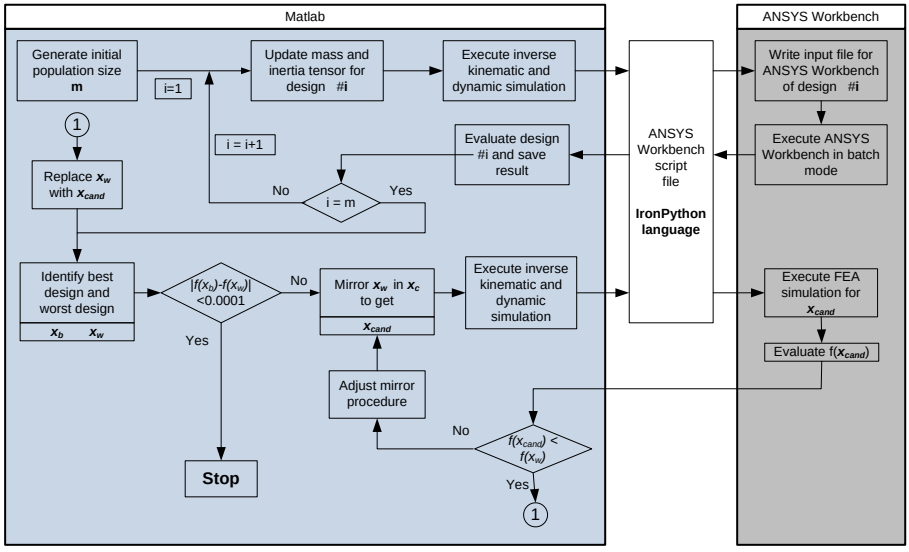


Figure 5.8: Diagram of the optimization routine.

5.8 The arm design optimization

An example of design is included to demonstrate the developed method. Prior to design optimizations, trajectories are defined for kinematic and dynamic analysis.

5.8.1 Arm Trajectories

To simplify the trajectory definition, straight-line motion is selected for the robotic arm. A straight line trajectory starting from an initial point $\mathbf{p}_0 = (x_0, y_0, z_0)^T$ at $t = 0$ to an ending point $\mathbf{p}_e = (x_e, y_e, z_e)^T$ at time $t = T$, can be expressed as

$$\mathbf{p} - \mathbf{p}_0 = u_p(\mathbf{p}_e - \mathbf{p}_0), \quad u_p \in [0, 1] \tag{5.25}$$

where, the parameter u_p controls the movement of the end-effector. A trajectory with $\mathcal{C}^2$-continuity can be planned as

$$u_p(t) = p_0 + p_1 t + p_2 t^2 + p_3 t^3 \tag{5.26}$$

where p_0, p_1, p_2 and p_3 are constant coefficients.

Assuming the velocities at the initial and ending point are $\dot{\mathbf{p}}_0 = (\dot{x}_0, \dot{y}_0, \dot{z}_0)^T$ and $\dot{\mathbf{p}}_e = (\dot{x}_e, \dot{y}_e, \dot{z}_e)^T$, the four constant coefficients can be solved by

$$p_0 = x_0;\ p_1 = \dot{x}_0 \tag{5.27a}$$

$$p_2 = \frac{3}{T^2}(x_e - x_0) - \frac{2}{T}\dot{x}_0 - \frac{1}{T}\dot{x}_e \tag{5.27b}$$

$$p_3 = -\frac{2}{T^3}(x_e - x_0) + \frac{1}{T^2}(\dot{x}_0 + \dot{x}_e) \tag{5.27c}$$

In this work, we use a group of four trajectories to conduct kinematics and dynamics simulation on the robotic arm, with the coordinates of the initial and end points of the trajectories listed in Table. 5.2. Among them, the end-effector moves horizontally following Trajectory 1, while moves vertically with Trajectory 2. Trajectories 3 and 4 are paths of different inclination. The robotic arm starts to move from rest and stops in five seconds. The Euler angles for the end-effector are given as $[0, \cos(t/20), 0]$, which implies the end-effector remains horizontal during the motion. In each iteration of

Table 5.2: Initial and end points of end-effector trajectories.

Trajectory	Initial point [mm]			End point [mm]		
	x_0	y_0	z_0	x_e	y_e	z_e
1	100	850	300	850	100	300
2	500	500	200	700	700	200
3	500	500	300	550	550	800
4	100	850	200	850	100	700

the optimization, the kinematics and dynamics are analyzed with respect to the four trajectories. The maximum torques of each joint are used to select motors and gearboxes for the drive train. Depending on the applications, the group of trajectories can be extended to contain more trajectories for more detailed evaluations of torque requirements of the robotic arm.

5.8.2 Material strength limits

The payload is defined as a point mass of $5kg$. In the finite element analysis, the design payload is multiplied by a safety factor, i.e. $F_A = 100$ [N]. The structure parts of this robot in this work are made of aluminium, so the yield strength $S_y = 280$ MPa. The deflection limit at the end-effector is set to $D_{lim} = 5$ mm.

5.8.3 Candidate components

Nine candidate motors from the Maxon Motor catalogue are considered. They are listed in a database ascendingly with respect to the mass of motor, as shown in Table 5.5 of Appendix. The gearboxes used in the robotic arm are selected from Harmonic Drive CPU units, as listed in Table 5.6 of Appendix. For the Harmonic Drive gearboxes, the efficiency is a function of operation speed. In this work, the gear efficiency is set to 0.85 for all gearboxes, which is an average value from product catalog.

The gear ratio of each joint is set to $\rho = \{200, 200, 200, 51, 100\}$, orderly from Joint 1 to Joint 5. Note there are two-stage gearboxes in Joints 1, 2 and 3, consisting of a planetary gearhead and a Harmonic Drive unit. For simplicity, only the mass of the Harmonic Drive gearbox is parameterized, while the mass of the planetary gearhead is set to constant. The Harmonic Drive CPU unit is adopted in all joints except Joint 4, due to the joint structure consideration. A planetary gearhead is used in Joint 4, so $u_{g,4} = 0$.

5.8.4 Optimization results

Optimized designs of structural dimensions and drive-train for the robotic arm are listed in Table 5.3. As shown in the optimization results of Case A, the optimized weight of the robotic arm is 8.3 kg, a mass reduction to 50% of the initial design being achieved.

Table 5.3: Results of design optimization.

Joint	Initial		Case A		Case B		Case C	
	Motor	Gearbox	Motor	Gearbox	Motor	Gearbox	Motor	Gearbox
1	RE 40	CPU 17	RE 30	CPU 14	EC 32	CPU 14	RE 35	CPU 14
2	RE 35	CPU 17	RE 25	CPU 14	RE 25	CPU 14	RE 25	CPU 14
3	RE 35	CPU 17	RE 30	CPU 14	RE 30	CPU 14	RE 35	CPU 14
4	RE 35	Gearhead	RE 25	Gearhead	RE 25	Gearhead	RE 25	Gearhead
5	RE 35	CPU 17	RE 25	CPU 14	RE 25	CPU 14	RE 25	CPU 14
Ratio	$r = 0.5$		$r = 0.6$		$r = 0.6$		$r = 0.5$	
Weight	16.7 [kg]		8.3 [kg]		9.92 [kg]		9.98 [kg]	

Case A: Optimization with the new method.
Case B: Optimization of drive-train with kinematic constraints.
Case C: Optimization of drive-train only [63].

The convergence of the objective function are depicted in Fig. 5.9, both the best value (black dot) and worst value (gray dot) from the Complex algorithm are shown. The solution to the optimal result is achieved at 6500 iterations with 150 population sizes. In this work, the tolerance of

convergence is equal to 0.0001.

The FEA in ANSYS Workbench is very computationally expensive. It takes 5 minutes for a single simulation. To increase the efficiency, FEA simulations were conducted in batch mode for the discrete structural dimensions and the results consisting of maximum stress, deformation and mass are stored in a database file. In each iteration of the optimization, the program will load the FEA results from the database for the integrated optimization instead of running FEA simulation. Adopting this approach leads to the computational time reduced from more than 10 days for one case to 10 minutes only.

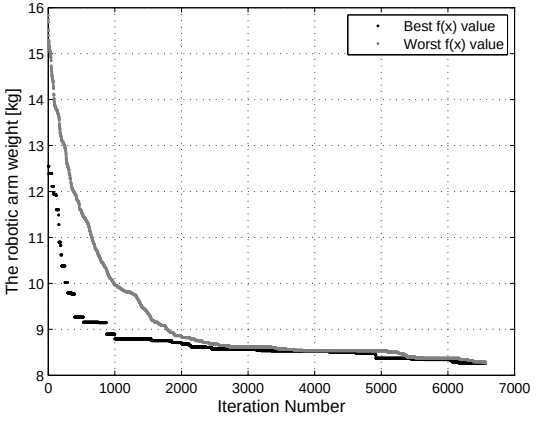
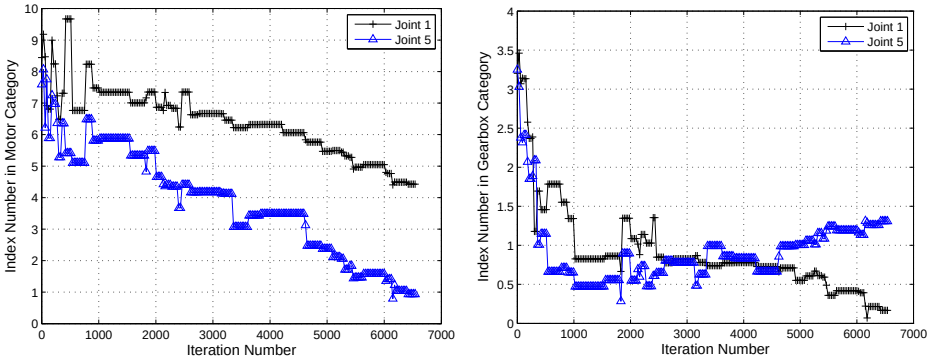


Figure 5.9: Convergence of the weight of the robotic arm.

Figure 5.10(a) illustrates the convergence of motor design variables. Only the convergence plots for Joints 1 and 5 are displayed for clarity. The convergence of gearbox design variables is depicted in Fig. 5.10(b). Comparing the convergence rate for the motor and gearbox design variables, the gearbox design variables converging rate towards the optimal results is faster than the motor design variables. This phenomena is caused by that the mass difference among Harmonic Drive units is larger than among motors.

The convergence of the link length ratio is shown in Fig. 5.11. The link length ratio is converged to $r = 0.6$. Comparing this ratio to some robotic manipulators [61] such as KUKA-KR-R650 ($r = 0.55$), Denso-VM-6083D series ($r = 0.54$), Mitsubishi-RV-2AJ series ($r = 0.61$) and Staubli-Tx40 series ($r = 0.58$), the optimal ratio agrees generally with these industrial robots. The difference between these ratios can be considered as the



(a) Motors (b) Gearboxes

Figure 5.10: Convergence plots for the design variables of motors and gearboxes.

influence on arm shape and mass distributions.

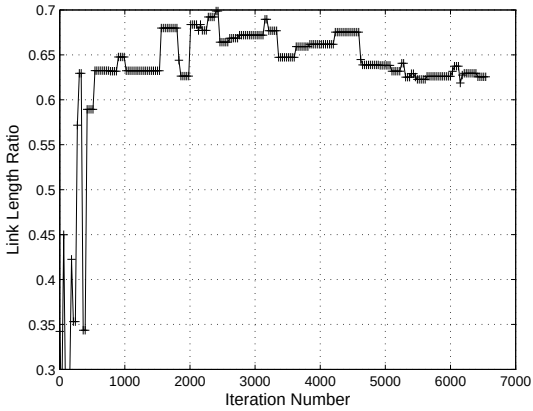


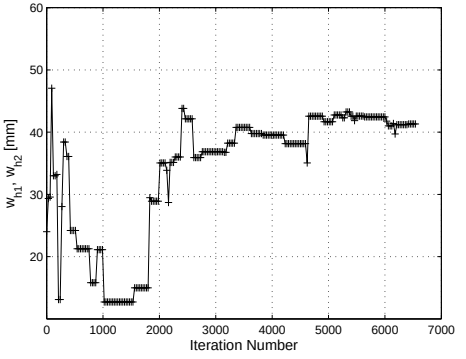
Figure 5.11: Convergence of the link length ratio.

The lengths of the upper arm link l_{s1} and lower arm link l_{s2} converge following the convergence of the link length ratio r. The optimized structural dimensions of the robotic arm are shown in Table 5.4. The convergence plots of the inner radius and widths of the opening slots are depicted in Fig. 5.12. Note that to reduce calculation, w_{h1} is made identical to w_{h2} in this work. According to the structural dimensions in Table 5.4, FEA is conducted separately for the original and optimized robotic arm designs, with the von-Mises element stress being depicted in Fig. 5.13.

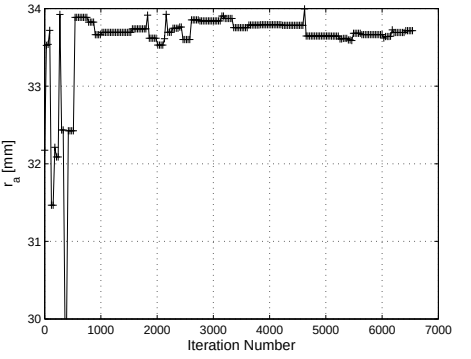
The variations of motor torques of Joints 1 and 2 for the initial and the

Table 5.4: Optimal structural dimensions for minimization of weight [mm].

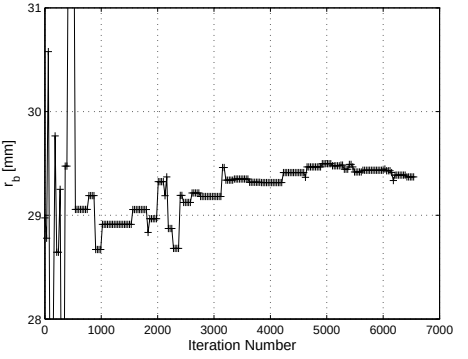
	l_1	r_a	r_b	w_{h1}	w_{h2}
Original	500	31	27	20	20
Optimized	600	34	29	40	40



(a) w_{h1} and w_{h2}



(b) Inner radius r_a



(c) Inner radius r_b

Figure 5.12: Convergence plots of structural dimensions.

optimal design of case 3 are shown in Fig. 5.14. The simulation is based on Trajectory 1 in Table 5.2. The torques of the optimal design are depicted in black, and that of the initial design are in gray. The RMS value of each torque is depicted with dashed line. It is seen that the optimal design has a reduction of 51% RMS torque for Joint 1, and a reduction of 72% RMS torque for Joint 2.

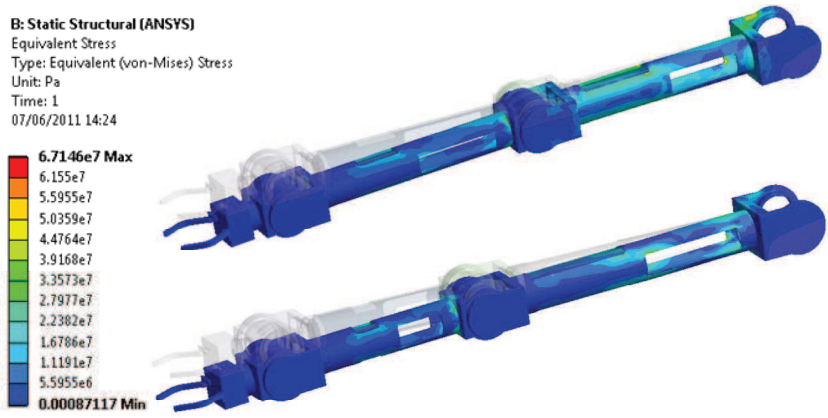
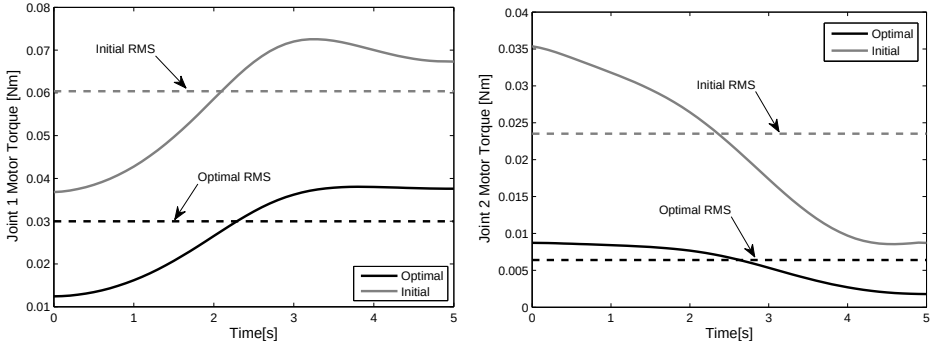


Figure 5.13: von-Mises element stress in the original (top) and optimized (bottom) robotic arms.



(a) Joint 1 (b) Joint 2

Figure 5.14: Motor torques for initial and optimal drive-train combinations.

The optimization results were compared with the results from a previous method [63]. Two additional cases are considered for comparison, one case with link lengths considered as design variables, and the other with all dimensions fixed. Results are summarized in Table 5.3. It is seen that the weight change of the robot is not significant in the two optimized cases without strength constraints, no matter the kinematic constraints are included or not. A major mass reduction is achieved with the optimization under the constraint of strength, which reduces the mass of the upper and lower arm links by 1.7 kg. The comparison reveals that the new method can contribute to reduce further the robot mass without degrading the

performance of the robot.

The prototype of the 5-dof robotic arm is shown in Fig. 5.15. The components of drive-train in the prototype are selected and scaled based on the optimization results shown in Table 5.3. The prototype doesn't utilize the optimized ratio of $r = 0.6$; instead, the length ratio is $r = 0.5$.

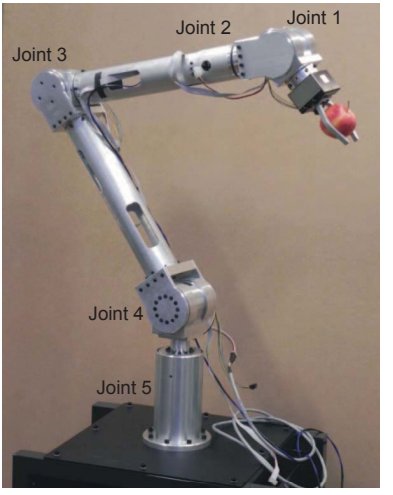


Figure 5.15: Prototype of the robot arm.

5.9 Discussion and conclusions

An integrated approach for the design of light-weight robotic arms was proposed in this work. Selections of structural dimensions, motors and gearboxes were formulated as a discrete optimization problem, which was solved by a non-gradient optimization method. Global conditioning index was taken as a constraint on kinematics performance of the robot. The results show that the method can achieve an optimal design with minimum mass, while satisfying the constraints on kinematics, drive-train and structural strength.

The inclusion of the robot structural strength in the optimization benefits the robot design in several aspects. Firstly, the mass can be effectively reduced by applying the static strength constraint, as did in this work. Secondly, this approach can also address the fatigue limit, a major concern in robot design, by either specifying a minimum stress or conducting fatigue

simulation in FEA module.

The proposed approach provides a systemic design optimization method for robots. For a draft robot design with given joint configurations, the approach can be used to select drive-train components and structural dimensions for lightweight purpose. Other design criteria like minimal cost can also be included and defined in the objective function in the proposed approach. Future work will include the generalization of this method for different objectives and also the integration of robot control into the optimization.

Appendix

Table 5.5: Candidate motor data from Maxon Motor [53].

Index No.	Maxon Motor	T_m [Nm]	T_m^{max} [Nm]	N_m^{max} [rpm]	J_m $[g \cdot cm^2]$	m_m [kg]
1	RE 25	0.0284	0.28	14000	10.5	0.13
2	RE 26	0.0321	0.227	14000	12.1	0.15
3	EC-i 40	0.0667	1.81	15000	24.2	0.21
4	RE 30	0.0882	1.02	12000	34.5	0.238
5	EC 32	0.0426	0.353	25000	20	0.27
6	RE 35	0.0965	0.967	12000	67.4	0.34
7	RE 36	0.0795	0.785	12000	67.2	0.35
8	EC 40	0.127	0.94	18000	85	0.39
9	RE 40	0.184	2.5	12000	138	0.48

Table 5.6: Candidate gearbox data from Harmonic Drive [54].

Index No.	Unit Size	T_g [Nm]	T_g^{max} [Nm]	N_g^{max} [rpm]	J_g $[kg \cdot m^2]$	m_g [kg]
1	14	11	54	8500	0.033×10^{-4}	0.54
2	17	39	110	7300	0.079×10^{-4}	0.79
3	20	49	147	6500	0.193×10^{-4}	1.3
4	25	108	284	5600	0.413×10^{-4}	1.95

6

Article IV

Modeling of Human Arm Energy Expenditure for Predicting Energy Optimal Trajectories

The paper has been published as:

Lelai Zhou, Shaoping Bai, Michael R. Hansen, and John Rasmussen, "Modeling of Human Arm Energy Expenditure for Predicting Energy Optimal Trajectories", *Modeling, Identification and Control*, vol. 32, no. 3, pp. 91-101, 2011. doi:10.4173/mic.2011.3.1

Abstract

Human arm motion can inspire the trajectory planning of anthropomorphic robotic arms to achieve energy-efficient movements. An approach for predicting metabolic cost in the planar human arm motion by means of the biomechanical simulation is proposed in this work. Two biomechanical models, including an analytical model and a musculoskeletal model, are developed to implement the proposed approach. The analytical model is developed by modifying a human muscle expenditure model, in which the muscles are grouped as torque providers for computation efficiency. In the musculoskeletal model, the predication of metabolic cost is conducted on the basis of individual muscles. With the proposed approach, metabolic costs for parameterized target-reaching arm motions are calculated and utilized to identify optimal arm trajectories.

6.1 Introduction

A human arm has seven dof (degrees-of-freedom) upon basic definition, three in the shoulder, two in the elbow, and two in the wrist. The redundancy in the arm dof implies infinite possible trajectories for a given movement task. For instance, when we pick up a bottle of water, there are a great number of trajectories that the arm can follow. With the hand located at a fixed point, the arm can also have different orientations.

The mechanism behind the selection of the predictable trajectory has been the subject of study over the years. The kinematic analysis [66, 67] revealed some interesting kinematic features of arm motions, but could not explain the planning mechanism for the activation of the individual muscle. One effective approach to study the planning mechanism is to examine the mechanical and physiological properties of a muscle, and to investigate the behaviour of individual muscles in human arm trajectories [68, 69, 70]. Experimental data on multi-joint human arm trajectories obtained from restricted horizontal planar movements have shown that human point-to-point arm motion trajectories have bell-shaped velocity profiles [71, 72]. Efforts were made to explain the observed trajectories as solutions to optimization problems. Optimization criteria have been proposed including minimum jerk theory [66], minimum travel cost theory [73], minimum isometric torque derivative [74], and averaged specific power [75]. The criteria used in the optimal trajectory study include also the minimum energy cost hypothesis for human arm trajectories presented and

tested by [76], among others.

This paper reports our study of human arm in planar motion. Our study focuses on the metabolic energy costs in human arm motions. Two human arm models, one analytical and one musculoskeletal, are proposed within our study. In the analytical model, the arm is represented by a 2-dof linkage driven by 4 torque providers (groups of muscles). The muscle metabolism model is modified from a human muscle energy expenditure model proposed by [77]. The musculoskeletal model is built in the AnyBody^{TM} Modeling System [78]. Both models are applied to planar arm motion in reaching a group of four targets. For each pair of target points, metabolic energy costs associated to parameterized arm trajectories are simulated and analyzed, from which the optimal arm trajectory for each pair of target points is further identified. The developed models are compared with a model reported in [76]. The analytical model is efficient in calculating energy consumptions and predicting optimal trajectories.

6.2 Model of metabolic cost in arm motion

6.2.1 Model of arm

A simplified arm model is shown in Fig. 6.1. Confined to planar motion, this arm has only two dof, with the shoulder joint situated at the origin of the coordinate system. The parameters of the arm model are measured or taken from [79], as listed in Table 6.1. Four target points P_1, P_2, P_3 and P_4 in Fig. 6.1 are specified, with their coordinates (in meter) being (0, 0.2), (0, 0.5), (-0.2, 0.3) and (0.2, 0.3), respectively. Out of these four points, four pairs of starting-end points are established to generate trajectories for arm motions. The combinations of the pairs of points are organized in eight groups, as listed in Table 6.2, with the arrow indicating the trajectory direction from the starting point to the ending point. Movements between two points from Groups A1 to D1 indicate forward human arm motions, while those from Groups A2 to D2 indicate backward motions.

In this arm model, four pieces of torque providers are defined, as shown in Fig. 6.1. The torque providers, numbered from 1 to 4, stand for the shoulder flexor, the elbow flexor, the shoulder extensor, and the elbow extensor, respectively.

It is assumed that the torque providers inherit the biomechanical properties of individual muscles. Their models can thus be developed by extending the existing muscle models with modifications. The torque providers defined in

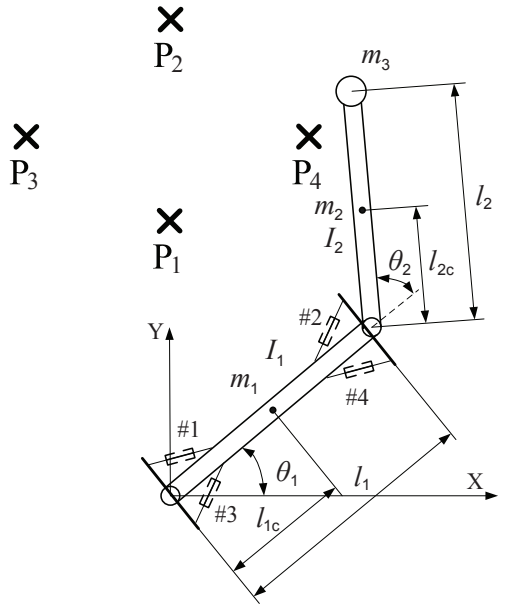


Figure 6.1: A simplified human arm model.

Table 6.1: Parameters of the human arm model.

Parameter	Value	Unit	Note
m_1	2.0	kg	upper arm mass
m_2	1.1	kg	lower arm mass
m_3	0.4	kg	hand mass
I_1	0.021	$kg \cdot m^2$	upper arm moment of inertia
I_2	0.007	$kg \cdot m^2$	lower arm moment of inertia
l_1	0.29	m	upper arm length
l_2	0.23	m	lower arm length
l_{1c}	0.14	m	distance from centre of mass to shoulder joint
l_{2c}	0.11	m	distance from centre of mass to shoulder joint

Table 6.2: Combinations of the target points.

Group A1	Group B1	Group C1	Group D1
$P_1 \rightarrow P_2$	$P_3 \rightarrow P_2$	$P_4 \rightarrow P_2$	$P_3 \rightarrow P_4$
Group A2	Group B2	Group C2	Group D2
$P_2 \rightarrow P_1$	$P_2 \rightarrow P_3$	$P_2 \rightarrow P_4$	$P_4 \rightarrow P_3$

this study adopt the Hill-type [80] muscle model, with parameters listed in Table 6.3, where the optimal fiber length L_{opt} of a torque provider comes from [81], while the maximum strength-moment M_{max} exerted on a joint by a torque provider is taken from [82].

Table 6.3: Parameters of the torque providers.

Torque Provider No.	L_{opt}[a] $[cm]$	M_{max}[b] $[Nm]$
♯1	16.2	92
♯2	17.3	77
♯3	27.9	67
♯4	13.4	46

[a] Optimal fiber length from [81]

[b] Muscle strength moment from [82]

6.2.2 Model of metabolic costs

The modeling of muscle metabolic costs is modified from the model of the human muscle energy expenditure proposed by [77]. Let $\dot{E}$ (in $Watt/kg$) be the total energy expenditure rate of a single muscle. It can be expressed as

$$\dot{E} = \dot{h}_a + \dot{h}_m + \dot{h}_{sl} + \dot{w}_{ce} \tag{6.1}$$

where $\dot{h}_a$ is the muscle activation heat rate, $\dot{h}_m$ is the maintenance heat rate, $\dot{h}_{sl}$ is the shortening/lengthening heat rate and $\dot{w}_{ce}$ is the mechanical power.

In this work, the muscle energy rate $\dot{E}$ is transformed into torque provider power P (in $Watt$) as a function of the joint moment and angular velocity, where both the shortening/lengthening heat rate and the mechanical power are included. Assuming that the length of a torque provider can never exceed L_{opt}, the metabolic power of a torque provider becomes

$$P = -M_{iso}\omega + G_s \frac{\rho}{\sigma} \frac{M_{max}}{L_{opt}} \omega \tag{6.2}$$

where the first term stands for the mechanical power and the second one stands for the heat rate due to shortening/lengthening. In Eq. (6.2), M_{iso} is the moment exerted on a joint by the muscle isometric force, and ω is the joint angular velocity. Moreover, σ is a specific tension, which takes a value of $0.25MPa$, as recommended by Umberger et al. The muscle density for mammalian muscle is $\rho = 1059.7\ kg \cdot m^{-3}$ [83].

The coefficient G_s (in N/kg) of Eq. (6.2) is a factor of the muscle shortening/lengthening heat rate [77], which depends on the percentage of fast twitch fibres (FT), and the shortening or lengthening of the muscle. When a muscle is shortening, the factor is calculated by

$$G_s = 0.0323 \times \%FT - 8.33 \tag{6.3}$$

As an example, when a muscle with 50% fast twitch fibres is shortening, $G_s = -6.72N/kg$; and when the muscle is lengthening, it takes the value $G_s = 33.3N/kg$.

The isometric moment M_{iso} is the moment exerted by the muscle. This moment can be obtained by modifying the mechanical joint moment with respect to the angular velocities of joints [84]. The isometric moment M_{iso} can be calculated as

$$M_{iso} = \begin{cases} \frac{M(1+Kr_\omega)}{1-r_\omega} & \text{for shortening} \\ \\ \frac{M(1-7.56Kr_\omega)}{1-0.8r_\omega-13.6Kr_\omega} & \text{for lengthening} \end{cases} \tag{6.4}$$

where $r_\omega = \omega/\omega_{max}$ with ω_{max} being the angular velocity corresponding to the maximum shortening speed of a muscle. ω_{max} is set as 22 rad/s for flexor, and 28 rad/s for extensor, according to [85]. The constant K depends on the muscle fibre type. Consequently, the metabolic power of a muscle becomes a function of joint angular velocity ω and joint moment M.

The joint mechanical moment can be calculated for shoulder and elbow joints, separately. In planar motion, the moment at the shoulder joint, M_s, is calculated as

$$\begin{aligned} M_s = {} & x_1 m_1 \ddot{y}_1 + x_2 m_2 \ddot{y}_2 + x_3 m_3 \ddot{y}_3 \\ & - y_1 m_1 \ddot{x}_1 - y_2 m_2 \ddot{x}_2 - y_3 m_3 \ddot{x}_3 \\ & + I_1 \dot{\omega}_1 + I_2 \left(\dot{\omega}_1 + \dot{\omega}_2\right) \end{aligned} \tag{6.5}$$

where (x_1, y_1), (x_2, y_2), and (x_3, y_3) specify the centres of mass of the upper arm, the lower arm, and the hand, respectively. The angular accelerations of the shoulder joint and the elbow joint are $\dot{\omega}_1$ and $\dot{\omega}_2$, respectively.

Likewise, the moment at the elbow joint, M_e, is given as

$$\begin{aligned} M_e = &(x_2 - x_e)\, m_2 \ddot{y}_2 + (x_3 - x_e)\, m_3 \ddot{y}_3 \\ &- (y_2 - y_e)\, m_2 \ddot{x}_2 - (y_3 - y_e)\, m_3 \ddot{x}_3 \\ &+ I_2\, (\dot{\omega}_1 + \dot{\omega}_2) \end{aligned} \tag{6.6}$$

where (x_e, y_e) are the coordinates of the elbow joint.

6.2.3 Parameterized arm motion

To describe all the possible arm motions, a Fourier series of joint angular velocities is considered

$$\omega = a_1 \sin\left(\frac{\pi t}{T}\right) + a_2 \sin\left(\frac{2\pi t}{T}\right) + a_3 \sin\left(\frac{3\pi t}{T}\right) + \cdots \tag{6.7}$$

where a_1, a_2, and a_3 are coefficients, T is the duration of the motion. Since the angular velocity has to be zero at $t = 0$ and $t = T$, only sine terms are included in this Fourier series form.

In our study, we use the first two sine terms of Fourier series to approximate angular velocity. Assuming θ_0 and θ_T be the angles of a certain joint at the $t = 0$ and $t = T$, the angular velocity can be expressed with two-term Fourier series as

$$\omega = \frac{\pi}{T}\left[\frac{\theta_T - \theta_0}{2} \sin\left(\frac{\pi t}{T}\right) + D \sin\left(\frac{2\pi t}{T}\right)\right] \tag{6.8}$$

where D is the deviation of a certain joint angle, following Alexander's definition [76]. Hereby, D_s denotes the shoulder angle deviation, and D_e the elbow angle deviation.

By changing the angle deviation factor D for both shoulder and elbow joints, different trajectories between the same pair of target points can be generated. Integrating both sides of Eq. (6.8) leads to

$$\theta = \frac{\theta_T - \theta_0}{2}\left[1 - \cos\left(\frac{\pi t}{T}\right)\right] + \frac{D}{2}\left[1 - \cos\left(\frac{2\pi t}{T}\right)\right] + \theta_0 \tag{6.9}$$

The angle of the shoulder θ_1 can vary from $-45°$ to $150°$, and the angle of the elbow θ_2 can vary from $0°$ to $150°$.

The joint angular acceleration is obtained by differentiating Eq. (6.8)

$$\dot{\omega} = \left(\frac{\pi}{T}\right)^2 \left[\frac{\theta_T - \theta_0}{2} \cos\left(\frac{\pi t}{T}\right) + 2D \cos\left(\frac{2\pi t}{T}\right)\right] \tag{6.10}$$

6.3 Musculoskeletal model

The torque providers used in the analytical model of the arm motion in Sec. 6.2 represent groups of muscles. It is desirable to extend the study to individual muscles and investigate the behavior of the muscles at the musculoskeletal level. To this end, we developed another model by taking advantage of a state-of-the-art biomechanical modeling system, namely, the AnyBody Modeling System [78].

A musculoskeletal right arm model was built in the AnyBody™ Modeling System, as shown in Fig. 6.2. The whole musculoskeletal model is comprised of 39 joints and 134 muscles. The model is derived from the repository models in AnyBody and each muscle unit is modeled using a three element Hill-type muscle model. In this study, as the model arm is confined to planar motion, only glenohumeral flexion joint and elbow flexion joint of the arm are free to move, and the others are constrained. In addition to the AnyBody model, a MatLab program was developed to control the changes of the parameters and trajectories.

Figure 6.2: A human musculoskeletal model built with the AnyBody Modeling System.

For human analytical musculoskeletal models, Hill-type muscle models are almost exclusively used. Hill-type muscle models consist of a contractile element (CE) that generates force and represents the muscle fibers, and a passive element (PE) in parallel with CE, and the above two modeling in series with a serial elastic element (T), as depicted in Fig. 6.3.

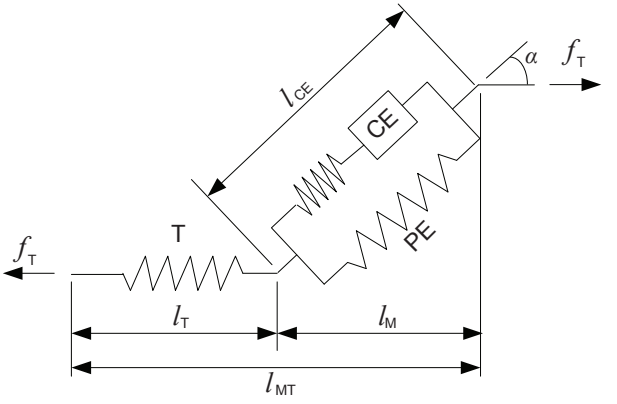


Figure 6.3: Hill-type muscle model consisting of three elements with parallel-series arrangement.

In the AnyBody modeling system, the mechanical power of an individual muscle-tendon unit is calculated by

$$P_{mech} = f_T v_{MT} = f_T \dot{l}_{MT} \tag{6.11}$$

The prediction of metabolic power is based on the efficiency of the contractile element in AnyBody as

$$P_m = \frac{P_{mech}}{\mu}, \quad \begin{cases} \mu = 0.25 & \text{for shortening} \\ \mu = -1.2 & \text{for lengthening} \end{cases} \tag{6.12}$$

6.4 The Alexander's model (reference model)

We compared our models with a model reported by [76], which is introduced briefly. The method of calculating metabolic power in Alexander's model made use of the work of [86]. The metabolic power of a uniarticular muscle is given by

$$P_{ref} = M_{iso}\omega_{max}\Phi(r_\omega) \tag{6.13}$$

where M_{iso}, ω_{max} and r_ω follow the definitions in Eq. (6.4). The data fit function Φ was adopted from Ma and Zahalak's work, with the form as

$$\Phi(r_\omega) = \begin{cases} 0.23 - 0.16\exp(-8r_\omega) & \omega \geq 0 \\ 0.01 - 0.11r_\omega + 0.06\exp(23r_\omega) & \omega < 0 \end{cases} \tag{6.14}$$

The Alexander's model consists of two antagonistic pairs of muscles to drive the arm. At any time, only one muscle of each pair is active. The Alexander's model uses mechanical power scaled by a function Φ as the main source of metabolic power for an individual muscle. While only two muscles are considered to be active, the reference model underestimates the metabolic power of human arm motion. In contrary, the analytical model we propose includes mechanical power and muscle shortening/lengthening heat together as the metabolic power. The analytical model utilizes torque providers working as muscle groups instead of only four muscles in the reference model. Our anatomical musculoskeletal model consists of 134 muscles, which is an extremely detailed model of the human arm.

6.5 Simulation routine

When running simulation with the analytical arm model, only one torque provider of each joint would be activated. The metabolic power of each torque provider is calculated using Eqs. (6.2)-(6.10). The metabolic energy of an individual torque provider is obtained by integrating the metabolic power over the entire duration. The metabolic costs of all individual torque providers are added together to obtain the total metabolic energy cost.

In the musculoskeletal arm model, all the muscles associated with arm motion are active. In each simulation, the program will write a file containing the joint angles. The program in Matlab runs the AnyBody console application in batch mode to conduct inverse dynamic analysis on the musculoskeletal arm. The metabolic powers of all the muscles are summed to obtain the overall metabolic power of the whole arm model. The metabolic cost can be obtained by integrating the overall metabolic power over the motion duration. The simulation routine of coupling Matlab with AnyBody is shown in Fig. 6.4.

For the analytical model, both the shoulder angle deviation D_s and the elbow angle deviation D_e are varied from $-40°$ to $40°$ in step size of $1°$. For the musculoskeletal model, D_s and D_e are varied from $-40°$ to $40°$ in step size of $5°$.

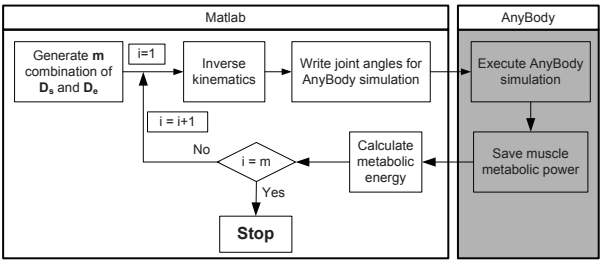


Figure 6.4: Simulation routine of the musculoskeletal model in the AnyBody software.

6.6 Simulation results

Biomechanical simulations are conducted on the two proposed models and the reference model as well. Simulation results of metabolic power, metabolic energy consumption, and optimal trajectories are compared to each other among the three models.

6.6.1 Predicted metabolic power

The metabolic powers predicted by the analytical, musculoskeletal and reference models are shown in Fig. 6.5. Four trajectories are selected from Groups A1 to D1, respectively. All trajectories are specified by $D_e = -20°$ and $D_s = 20°$.

There is barely experimental data about metabolic costs in human arm motion, due to lack of experimental techniques. We cannot compare the simulation results to the experiments to validate the model in a holistic level. However, there is much literature about experimental results and models on a single muscle [77, 86]. Since we can simulate the actions of all the muscles in a musculoskeletal arm, the whole metabolic cost for an arm can be calculated as well. The musculoskeletal model is a much detailed human arm model with 134 muscles, which means that the musculoskeletal model predicts the metabolic power consumptions as the sum of the metabolic power of each individual muscle.

In general, the analytical model predicts metabolic power similar to the musculoskeletal model. There is a good agreement between the analytical and musculoskeletal models in the rates of change, referring to the increasing and declining parts of the metabolic power curves in Fig. 6.5. The reference model predicts similar metabolic power profile to the analytical model, but

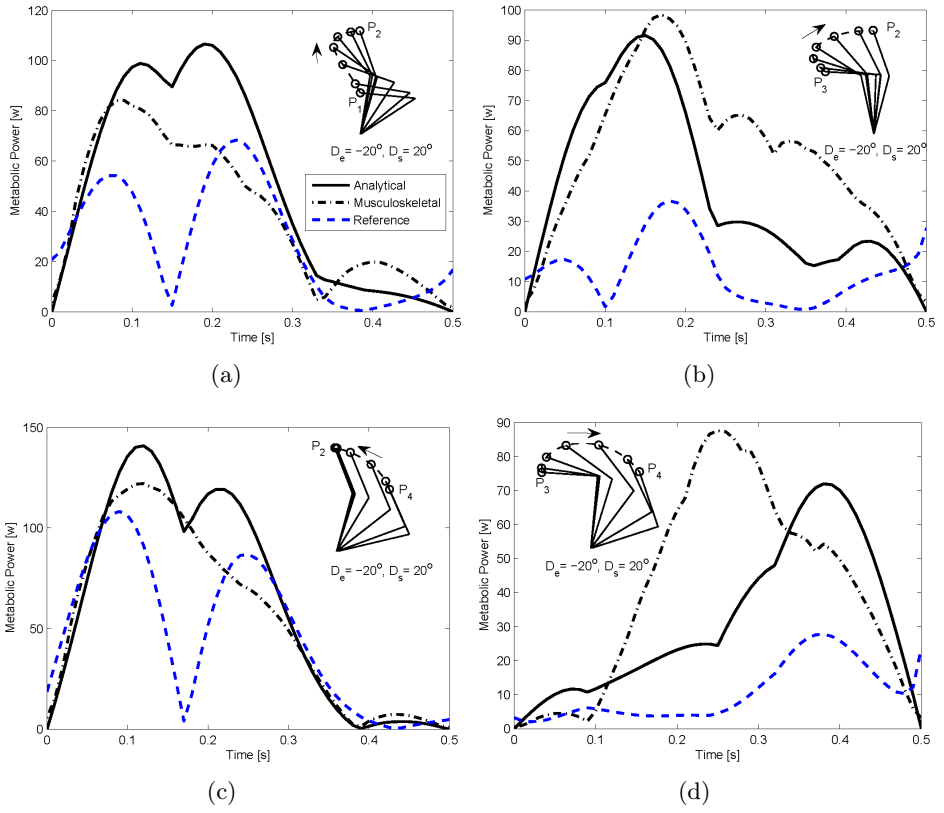


Figure 6.5: Metabolic powers of arm motion with four trajectories.

the maximum values are far below the two proposed models. The analytical model is more close to the musculoskeletal model in predicting metabolic power compared to the reference model.

6.6.2 Predicted metabolic energy

The metabolic energy cost of the analytical model for the parameterized trajectories in Group A1 is shown in Fig. 6.6(a), while those of the musculoskeletal model and the reference model [76] are shown in Fig. 6.6(b) and 6.6(c), respectively.

The duration of movements is 0.5s. In Fig. 6.6, each point (D_e, D_s) represents a trajectory. The global minimum is marked by a red cross. For the analytical model moving in Group A1, the minimum energy cost is found at $D_e = -7°$ and $D_s = 0°$, with a minimal value of $17.662J$. For

the musculoskeletal model moving in Group A1, the minimum is found at $D_e = 0°$ and $D_s = 10°$, with a minimal metabolic energy cost of $14.764J$. The CPU time for calculating minimal energy cost for the three models is list in Table 6.4. It is seen that the analytical model developed in this work is more efficient compared with the other two models.

Table 6.4: Computational time for the three models.

Model	Elapsed time [s]
Analytical	20
Musculoskeletal	7.13×10^4
Reference	37

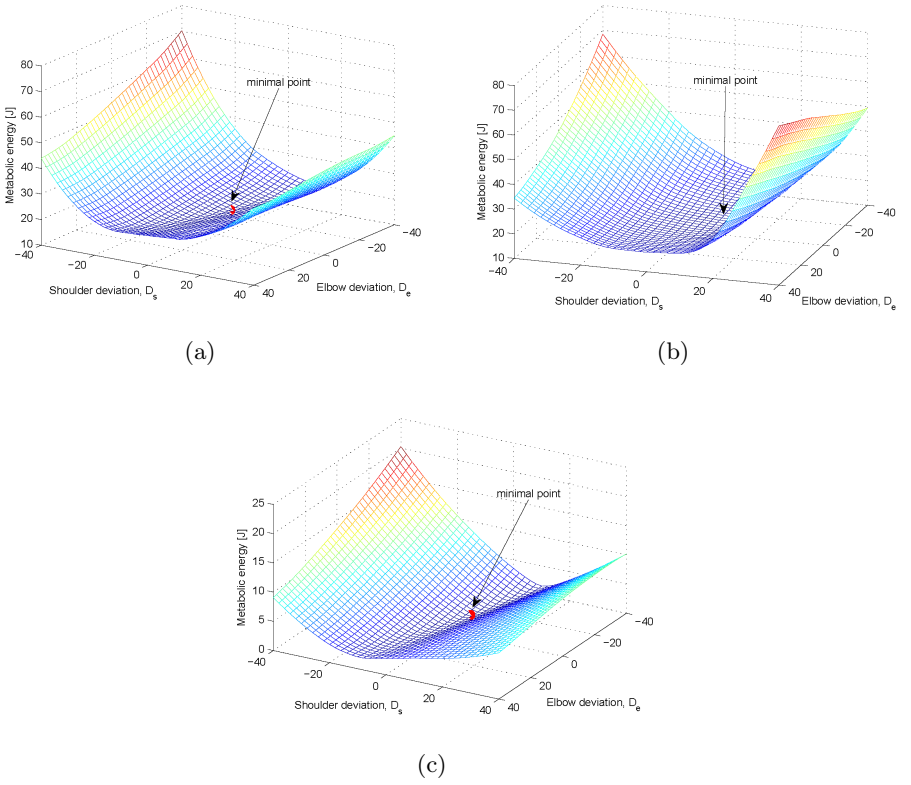
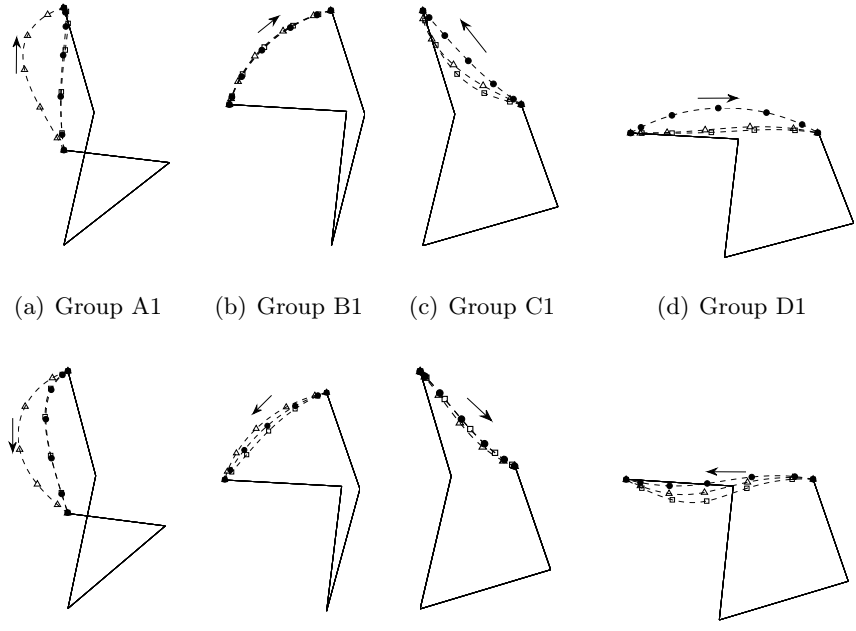


Figure 6.6: Metabolic energy cost contours obtained with different models, (a) the analytical model, (b) the musculoskeletal model, (c) the reference model reported in [76].

6.6.3 Optimal trajectories

Based on the metabolic energy cost contour, the optimal trajectories can be identified. The optimal trajectories predicted by the proposed two models and the reference model are shown in Fig. 6.7 for comparison. The optimal trajectories obtained with different models are marked with specified symbols. Optimal trajectories for both forward and backward arm motions are displayed. The dashed line indicates the path of the hand, and the moving direction is marked by the arrow. The elbow (D_e) and shoulder (D_s) angle deviations and metabolic energy costs (E_m) for optimal trajectories in Fig. 6.7 are summarized in Table 6.5. The elbow and shoulder joint velocities for the optimal trajectories from Groups A1 to D1 are depicted in Fig. 6.8, while Fig. 6.9 shows the corresponding hand velocities.



(a) Group A1 (b) Group B1 (c) Group C1 (d) Group D1

(e) Group A2 (f) Group B2 (g) Group C2 (h) Group D2

Figure 6.7: Optimal trajectories predicted by the proposed two models and a reference model [76]. The hand paths are marked with dots (•), triangle (△), and box symbols for the proposed analytical, musculoskeletal and reference models, respectively.

The two proposed models predict very similar optimal trajectories with respect to the reference model, even though the reference model greatly underestimates the metabolic energy consumption. There are some discrepancies of the optimal trajectories predicted by the musculoskeletal model and the other two models in Groups A1 and A2. This may be due to the deviations of the musculoskeletal model.

The hand velocities of the optimal trajectories show bell-shape profiles, which have a good agreement with the straight arm motion observed by [66]. Fig. 6.7 shows that unconstrained point-to-point motions are not only approximately straight, but also metabolic energy optimal.

The optimal trajectories in forward and backward arm movements, displayed in Fig. 6.7(a)-6.7(c) and Fig. 6.7(e)-6.7(g), respectively, show that they do not necessarily follow the same hand paths in forward and backward movements.

Table 6.5: Elbow (D_e) and shoulder angle deviation (D_s) and metabolic cost (E_m) of optimal trajectories in Fig.6.7.

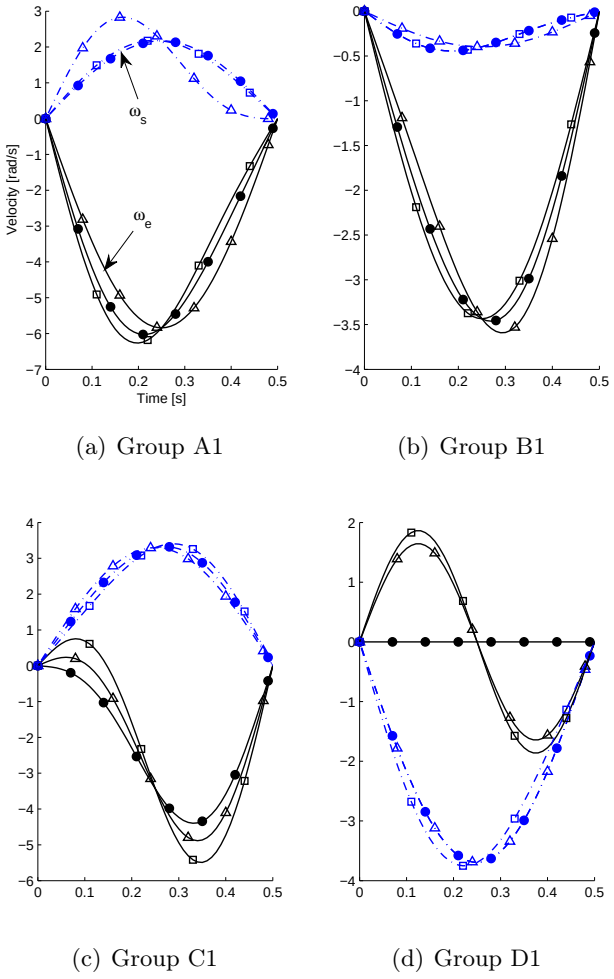
Model	Group A1			Group B1			Group C1			Group D1		
	D_e	D_s	E_m	D_e	D_s	E_m	D_e	D_s	E_m	D_e	D_s	E_m
Analytical	−7°	0°	17.662 J	2°	−1°	6.816 J	15°	−2°	22.934 J	0°	0°	6.939 J
Musculoskeletal	0°	10°	14.764 J	5°	0°	3.537 J	20°	0°	24.594 J	15°	0°	14.505 J
Ref. Model	−11°	1°	2.614 J	0°	−1°	1.46 J	26°	−4°	10.929 J	17°	−3°	1.422 J
	Group A2			Group B2			Group C2			Group D2		
Analytical	−1°	5°	28.792 J	5°	−3°	20.980 J	17°	−4°	19.982 J	20°	0°	23.751 J
Musculoskeletal	15°	10°	14.429 J	5°	0°	5.875 J	20°	−5°	14.670 J	25°	0°	26.340 J
Ref. Model	−3°	6°	3.916 J	10°	−5°	4.862 J	17°	−4°	2.045 J	30°	−2°	12.113 J

6.6.4 Metabolic power of optimal trajectories

Metabolic power of the optimal trajectories for target points in Groups A1 and C1 are calculated by the two proposed models and the reference model, as shown in Fig. 6.10. The metabolic power is calculated based on the optimal trajectories obtained by the analytical model. It is seen that the analytical and the musculoskeletal model predict similar profile of metabolic power consumption, even though the analytical model overestimates the power a bit than the musculoskeletal one. Nevertheless, the model reported by Alexander underestimates the metabolic power significantly compared to the two proposed models.

6.6.5 Different durations with the analytical model

Optimal trajectories in Group A1 for four different durations (0.3 s, 0.5 s, 0.8 s, 1s) predicted by the analytical model are shown in Fig. 6.11. Details of the optimal trajectories with the four time durations are listed in Table 6.6.



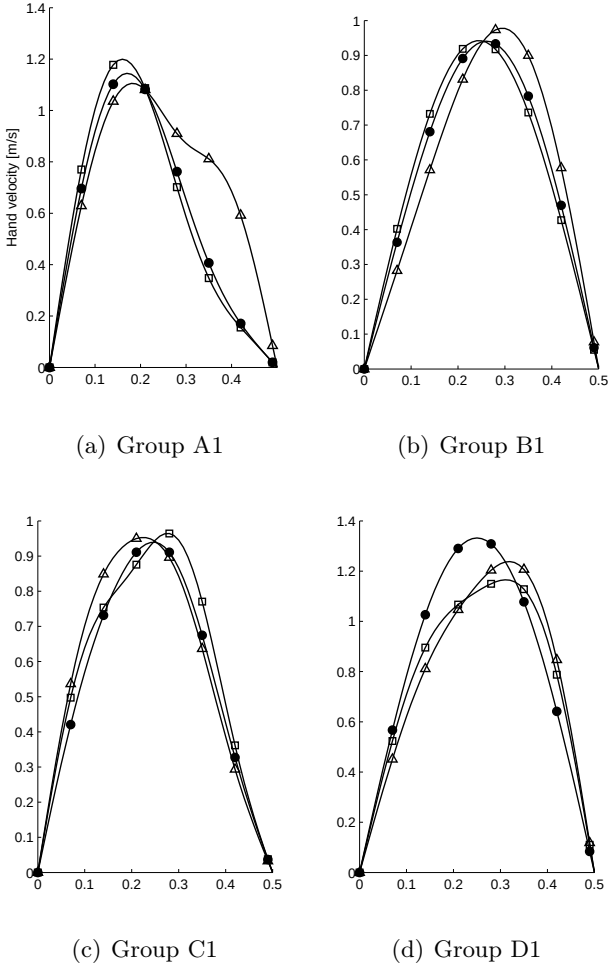
(a) Group A1 (b) Group B1

(c) Group C1 (d) Group D1

Figure 6.8: Angular velocities of elbow joint (ω_e, solid curve) and shoulder joint (ω_s, dashed curve) corresponding to optimal trajectories for motions of Groups A1, B1, C1 and D1 in Fig. 6.7. The velocity curves are marked with dots (•), triangle (△), and box symbols for the proposed analytical, musculoskeletal and reference models, respectively.

For motions in Groups B1, C1 and D1, all the optimal trajectories for four different durations are almost identical, according to the values of D_e and D_s in Table 6.6.

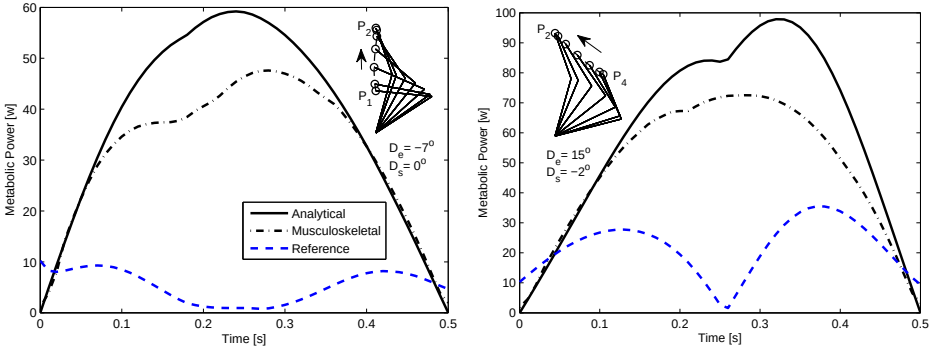
In Group A1, the optimal trajectories of the fast movements (0.3 s, 0.5 s) and those of the slow movements (0.8s, 1s) show a nearly straight path of



(a) Group A1 (b) Group B1

(c) Group C1 (d) Group D1

Figure 6.9: Hand velocities of the corresponding optimal trajectories of Groups A1, B1, C1 and D1 in Fig. 6.7. The velocity curves are marked with dots (•), triangle (△), and box symbols for the proposed analytical, musculoskeletal and reference models, respectively.



(a) Group A1 (b) Group C1

Figure 6.10: Metabolic power of optimal trajectories in Groups A1 and C1 predicted by the analytical, musculoskeletal and reference models.

Table 6.6: Elbow and shoulder angle deviations of optimal trajectories with different durations predicted by the analytical model.

Points	0.3 s		0.5 s		0.8 s		1 s	
	D_e	D_s	D_e	D_s	D_e	D_s	D_e	D_s
Group A1	−7°	0°	−7°	0°	−3°	−1°	24°	−12°
Group B1	2°	−1°	2°	−1°	2°	−1°	2°	−1°
Group C1	16°	−3°	15°	−2°	15°	−2°	15°	−2°
Group D1	0°	0°	0°	0°	0°	0°	0°	0°

the hand, as that [66] observed. No significant differences are observed.

6.7 Discussions

It can be noticed that all models, the analytical and musculoskeletal models and the reference model, lead to very similar optimal trajectories. Also, the model-predicated optimal trajectories match the observed human planar arm movements [66].

The analytical model takes a simple form with only four torque providers included, compared to the musculoskeletal model using 134 muscles. On the other hand, the metabolic energy costs calculated by the two proposed models and reference model have considerable differences, as shown in Table 6.5. In some cases, the muscles in a human arm are antagonistic to causing substantial energy waste. This part of energy cost is considered in the proposed analytical model and the musculoskeletal model, but not in

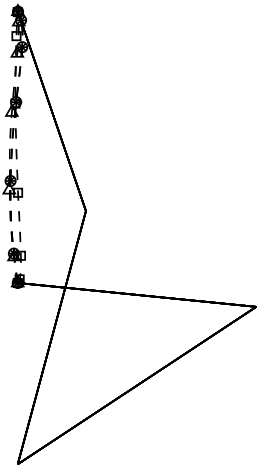


Figure 6.11: Optimal trajectories in Group A1 corresponding to four different durations by the proposed analytical model. The hand paths are marked with dots (∘), cross (×), triangle (△), and box symbols for duration 0.3 s, 0.5 s, 0.8 s, and 1 s, respectively.

the reference model. It is apparent that the metabolic costs calculated by the two models developed presently are much larger than that calculated by the reference model. In some cases, the metabolic cost calculated by the analytical model is even a bit larger than that by the musculoskeletal model. It is reasonable that the musculoskeletal model is considered as a better predictor of metabolic energy cost.

Predicted optimal trajectories in all groups are similar among the three models, except Groups A1 and A2. The differences are apparent if we look into the elbow angle deviation D_e and shoulder angle deviation D_s in Table 6.5. The elbow angle deviation D_e differences among Groups B1, C1, D1 is below 17°, and the shoulder angle deviation D_s differences is below 4°. Comparing forward and backward arm motions like Groups B1 and B2, the optimal trajectories of the forward motions (Group B1) are similar to those of the backward motions (Group B2). However, the optimal trajectories of forward motions in Group A1 and backward motions in Group A2 are different, especially those predicted by the analytical and reference models. The trajectories predicted by the two models in Fig. 6.7(a) show almost

straight paths, and those in Fig. 6.7(e) show apparent curved paths.

Besides, the computational times with the analytical and musculoskeletal models are significantly different. The analytical model involves only numerical calculation, and is much more efficient than the musculoskeletal model. A natural progression of the work is to investigate whether a joint moment-driven model, which can be even more computationally efficient, can also provide valid results, thus eliminating the need for muscles in the model.

6.8 Conclusions

In this paper, an analytical and a musculoskeletal models were developed and compared with a reference model. Both models are able to calculate metabolic cost. Optimal trajectories were identified with the developed models for planar arm movements. The comparisons between the three models showed that even though the two proposed models and the reference model predict similar optimal trajectories, the reference model greatly underestimates the metabolic cost. Among the two newly proposed models, the musculoskeletal model is more accurate in predicting the metabolic energy cost while the analytical one is more efficient in predicting optimal trajectories. Given the fact that the predication results from two models are very close, the simple analytical model can replace the complicated musculoskeletal model in predicting metabolic costs and energy efficient trajectories in certain applications where the overall behavior of the arm motion is of interest.

7

Conclusions

The main scope of this work is the development of a novel optimization approach for the design of lightweight robotic arms. A new optimization approach was developed for robot optimization to handle selection of motors and gearboxes, geometric and structural dimensions. This was achieved through stepwise optimization in three levels, starting from the constraints of motors and gearboxes, then the constraints of kinematic performance, and finally structural strength constraints.

7.1 Summary of articles

Article I

Article I presents an optimization method for robotic drive-trains. The selection of motors and gearboxes was formulated as a discrete optimization problem, which was solved by a non-gradient optimization algorithm. Constraints were formulated by considering both motor and gearbox characteristics and robotic arm dynamics. The proposed method is able to reach a design with lower mass for a given set of driving components. A co-simulation platform consisting of a MSC.ADAMS dynamics model and an optimization algorithm implemented in Matlab code was developed, which enabled design optimization based on dynamics of an embodiment created in CAD systems.

The total weight of the robotic arm was optimized to 10.2 kg, a reduction of 38% compared with the initial design. The solution to the optimal result was achieved after 3160 iterations with a population size of 140. The optimal design reduced the peak torque by 31.8% reduction for Joint 1 and by 40% for Joint 2.

Two ways of dealing with the design variables were investigated in order

to confirm a more efficient one. The linear design variable method yielded better results at the cost of more iterations and objective function evaluations. The rounded design variable method was more efficient according to the comparison results.

Article II

Article II reports an extension of the work in Article I by integrating the geometric dimension together with drive-train optimization. Global conditioning index (GCI) was taken as a constraint on kinematic performance of the robot. The robot dynamics was constrained by considering characteristics of motors and gearboxes.

With this method, the optimized mass of the robotic arm was equal to 9.92 kg, with a reduction of 41% corresponding to the initial combinations. The optimal link length ratio was found as $r = 0.6$. The solution to the optimal result was achieved at 3500 iterations with 130 population sizes. The optimized design had a reduction of 41.29% RMS torque for Joint 1, and a reduction of 26.87% RMS torque for Joint 2.

Article III

Article III presents an integrated approach for the design of lightweight robotic arms. Selections of structural dimensions, geometric dimensions, motors and gearboxes were formulated as a discrete optimization problem. Both structural strengths and end-effector deformation were formulated as constraints through finite element analysis. A co-simulation platform was developed, which integrated Matlab with ANSYS Workbench.

A major mass reduction was achieved with the optimization under the constraint of structural strength, which reduced further the mass of the upper and lower arm links by 1.7 kg. The integrated method yielded an optimized robotic arm of 8.3 kg, a mass reduction to 50% of the initial design. The solution to the optimal result was achieved at 6500 iterations with 150 population sizes. The optimal design had a reduction of 51% RMS torque for Joint 1, and a reduction of 72% RMS torque for Joint 2.

The optimization results from Article I to III are summarized in Table 7.1.

Article IV

Table 7.1: Summary of the robotic optimization results.

Article	Optimized Mass	Mass[1] Reduction	Torque[2] Reduction	Population Size	Iteration Number
I	10.2 kg	38%	31.8%	140	3160
II	9.92 kg	41%	41.29%	130	3500
III	8.3 kg	50%	51%	150	6500

[1] Relative to the mass of 16.7 kg of the initial design

[2] For Joint 1

Article IV presents preliminary study of human arm motion for energy optimal trajectories. An approach for predicting metabolic cost in the planar human arm motion by means of the biomechanical simulation was proposed in this work. Two biomechanical models, including an analytical model and a musculoskeletal model, were developed to implement the proposed approach. The analytical model was developed by modifying a human muscle expenditure model, in which the muscles were grouped as torque providers for computation efficiency. In the musculoskeletal model, the predication of metabolic cost was conducted on the basis of individual muscles. The simulation results of the two proposed models were compared to a reference model.

It showed from the simulation results that the analytical model predicted metabolic power similar to the musculoskeletal model. The two proposed models predicted very similar optimal trajectories with respect to the reference model, even though the reference model greatly underestimated the metabolic energy consumption. The simple analytical model could replace the complicated musculoskeletal model in predicting metabolic costs and energy efficient trajectories in certain applications where the overall behavior of the arm motion is of interest.

7.2 Concluding remarks

Through utilizing optimization method in robot design, a fully integrated optimization approach was developed for the design of lightweight robots. The approach provides a systematic robot design optimization method for lightweight purpose. For a draft robot design with given joint configurations, the approach can be used to select drive-train components, geometric and structural dimensions. Other design criteria like minimum

cost, minimum energy consumption could also be included and defined in the objective function in the proposed approach.

A 5-dof lightweight robotic arm was designed and developed based on the selected drive-train components and dimensions from the optimization. The prototype weighs 14 kg with a payload capacity of 5 kg.

The optimization approach in this thesis presents a structured and efficient way of addressing a robot design problem where much insight could be gained during the iterative process of solving the optimization problem.

7.3 Contributions

Within this project, the following contributions to the design and optimization of lightweight robotic arms were made

- New robotic optimization methods were developed. It is the first time to integrate the drive-train, kinematics and structural dimensions together in the optimization design of robots for minimal mass.
- Three extensible simulation platforms for robot simulation were developed. The platforms integrate numeric programming software with commercial dynamic simulation and FEA simulation software. The platforms could be easily expanded to contain more design variables on different robotic parameters and the corresponding constraints.
- A prototype of the 5-dof lightweight robotic arm was built to validate the optimization approaches. The prototype can be used to validate the different simulation models developed within the project.
- Two new human arm motion models were proposed for energy-optimal trajectory predictions. The models provide an approach of studying arm control strategies of generating arm trajectories.

7.4 Future work

The optimization approach in this thesis focused on the mechatronic part of the robotic arm. Robot control is a key competence for robot manufacturers and is very important in order to get as much performance as possible out of a robot. Tuning of control parameters is also crucial for a robotic arm. One possible direction of the future work is to combine the mechanical

system design together with the control system design in the whole system optimization. Control parameters could be taken as design variables in the optimization.

A 5-dof lightweight robotic arm was built in the project. A control system has been developed for the robot. One task is to develop a control model containing control methods for trajectory planning. Preliminary study has been conducted on the control strategies of human arm through biomechanics modeling and simulation. More studies are needed to implement the control strategies and develop a control model for the robot.

Bibliography

[1] O. Ivlev, C. Martens, and A. Graeser. Rehabilitation robots FRIEND-I and FRIEND-II with the dexterous lightweight manipulator. *Technology and Disability*, 17:111–123, 2005.

[2] Z. Bien, M. J. Chung, P. H. Chang, and D. S. Kwon. Integration of a rehabilitation robotic system (KARES II) with human-friendly man-machine interaction units. *Autonomous Robots*, 16:165–191, 2004.

[3] R. M. Mahoney. The raptor wheelchair robot system. In M. Mokhtari, editor, *Integration of Assistive Technology in the Information Age*, pages 135–141. IOS press, 2001.

[4] A. Albu-Schäffer, S. Haddadin, C. Ott, A. Stemmer, T. Wimböck, and G. Hirzinger. The DLR lightweight robot: design and control concepts for robots in human environments. *Industrial Robot*, 34(5):376–385, 2007.

[5] JACO. Kinova technology, 2011. Available from: `http://www.kinovatechnology.com/`.

[6] A. Jardon, A.Gimenez, R. Correal, R. Cabas, S.Martinez, and C.Balaguer. A portable light-weight climbing robot for personal assistance applications. *Industrial Robot*, 33(4):303–307, 2006.

[7] H. H. Kwee. Integrated control of MANUS manipulator and wheel chair enhanced by environmental docking. *Robotica*, 16(5):491–498, 1998.

[8] B. Rooks. The harmonious robot. *Industrial Robot*, 33(2):125–130, 2006.

[9] A. Ananiev, D. Ignatova, and B. Iliev. An approach to the design of a light-weight reconfigurable robot arm for a mobile robot. *Problems of Engineering Cybernetics and Robotics*, 53:93–100, 2002.

[10] Y. Ogura, H. Aikawa, Kazushi, H.-O. Lim, and A. Takanishi. Development of a new humanoid robot wabian-2. In *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, pages 76–81, 2006.

[11] ASIMO. Asimo website, 2011. Available from: `http://world.honda.com/ASIMO/`.

[12] A. Albers, S. Brudniok, J. Ottnad, C. Sauter, and K. Sedchaicharn. Upper body of a new humanoid robot - the design of armar iii. In *Proceedings of the 2006 IEEE-RAS International Conference on Humanoid Robots*, pages 308–313, 2006.

[13] A. Edsinger-Gonzales and J. Weber. Domo: a force sensing humanoid robot for manipulation research. In *Proceedings of the 2004 IEEE-RAS International Conference on Humanoid Robots*, pages 273–291, 2004.

[14] F. Roos, H. Johansson, and J. Wikander. Optimal selection of motor and gearhead in mechatronic applications. *Mechatronics*, 16(1):63–72, 2006.

[15] K. Pasch and W. Seering. On the drive systems for high-performance machines. *ASME Journal of Mechanisms*, 106:102–108, 1983.

[16] H. J. Van de Straete, J. de Schutter, P. Degezelle, and R. Belmans. Servo motor selection criterion for mechatronic applications. *IEEE Transactions on Mechatronics*, 3(1):43–50, 1998.

[17] H. J. Van de Straete, J. de Schutter, and R. Belmans. An efficient procedure for checking performance limits in servo drive selection and optimization. *IEEE Transactions on Mechatronics*, 4(4):378–386, 1999.

[18] P. Chedmail and M. Gautier. Optimum choice of robot actuators. *Journal of Engineering for Industry*, 112:361–367, 1990.

[19] M. Pettersson and J. Ölvander. Drive train optimization for industrial robots. *IEEE Transactions on Robotics*, 25(6):1419–1423, 2009.

[20] H. Elmqvist, H. Olsson, S. E. Mattsson, and D. Brück. Optimization for design and parameter estimation. In *Proceedings of the 4th International Modelica Conference*, pages 255–266, 2005.

[21] A. Bowling and O. Khatib. Dynamic loading criteria in actuator selection for desired dynamic performance. *Advanced Robotics*, 17(7):641–656, 2003.

[22] D. Z. Chen. Drive train configuration arrangement for gear coupled manipulators. *Journal of Robotic Systems*, 14(8):601–612, 1997.

[23] Y. Zhu, J. Qiu, and J. Tani. Simultaneous optimization of a two-link flexible robot arm. *Journal of Robotic Systems*, 18(1):29–38, 2001.

[24] B. K. Rout and R. K. Mittal. Optimal design of manipulator parameter using evolutionary optimization techniques. *Robotica*, 28:381–395, 2010.

[25] P. S. Shiakolas, D. Koladiya, and J. Kebrle. Optimum robot design based on task specifications using evolutionary techniques and kinematic, dynamic, and structural constraints. *Inverse Problems in Science and Engineering*, 10(4):359–375, 2010.

[26] Z. Shiller and S. Sundar. Design of robotic manipulators for optimal dynamic performance. In *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, pages 344–349, 1991.

[27] O. Khatib and J. Burdick. Optimization of dynamics in manipulator design: the operational space for formulation. *International Journal of Robotics and Automation*, 2(2):90–98, 1987.

[28] A. Albers, J. Ottnad, H. Weiler, and P. Haeussler. Methods for lightweight design of mechanical components in humanoid robots. In *Proceedings of the 7th IEEE-RAS International Conference on Humanoid Robots*, pages 609–615, 2007.

[29] J. Roy, R. P. Goldberg, and L. L. Whitcomb. Structural design, analysis, and performance evaluation of a new semi-direct drive robot arm: Theory and experiment. *IEEE/ASME Transactions on Mechatronics*, 9(1):10–19, 2004.

[30] A. Pil and H. Asada. Rapid recursive structure redesign for improved dynamics of a single link robot. *ASME Journal of Dynamic Systems, Measurement, and Control*, 117(4):520–526, 1995.

[31] M. J. Box. A new method of constrained optimization and a comparison with other methods. *Computer Journal*, 8:42–52, 1965.

[32] H. J. Holland. *Adaptation in natural and artificial systems, an introductory analysis with application to biology, control and artificial intelligence.* MIT press, 1992.

[33] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220:671–680, 1983.

[34] F. Glover. Tabu search - part i. *ORSA Journal on Computing*, 1:190–206, 1989.

[35] D. Marvis and S. Qiu. An improved process for the generation of drag polars for use in conceptual/preliminary design. In *Proceedings of SAE World Aviation Conference*, San Francisco, USA, October 1999.

[36] H. A. Simon. *Models of bounded rationality.* Cambridge Univ. Press, Cambridge, MA, 1982.

[37] L. Borup and A. Parkinson. Comparison of four non-derivative optimization methods on two problems containing heuristic and analytic knowledge. In *Proceedings of ASME Advances in Design Automation*, Scottsdale, Arizona, 1992.

[38] J. Denavit and R. S. Hartenberg. A kinematic notation for lower pair mechanisms based on matrices. *ASME J. Appl. Mech.*, 77:215–221, 1955.

[39] L. W. Tsai. *Robot analysis: The mechanics of serial and parallel manipulators.* John Wiley & Sons, 1999.

[40] S. Cetinkunt. Optimal design issues in high-speed high-precision motion servo systems. *Mechatronics*, 1:187–201, 1991.

[41] G. Cusimano. A procedure for a suitable selection of laws of motion and electric drive systems under inertial loads. *Mechanism and Machine Theory*, 38:519–553, 2003.

[42] G. Cusimano. Optimization of the choice of the system electric drive-device–transmission for mechatronic applications. *Mechanism and Machine Theory*, 42:48–65, 2007.

[43] C. Leger. *Automated synthesis and optimization of robot configurations: an evolutionary approach.* PhD thesis, Carnegie Mellon University, Pittsburgh, USA, 1999.

[44] T. Izumi, H. Zhou, and Z. Li. Optimal design of gear ratios and offset for energy conservation of an articulated manipulator. *IEEE Transactions on Automation Science and Engineering*, 6(3):551–557, 2009.

[45] D.-Z. Chen and S.-C. Wang. On the drive train design of gear coupled manipulators. *Journal of Robotic Systems*, 15(8):477–486, 1998.

[46] L. Zhou, S. Bai, and M. R. Hansen. Design and kinematics of a 5-dof light-weight anthropomorphic robotic arm. In *Proceedings of the 22nd Nordic Seminar on Computational Mechanics*, pages 205–208, Aalborg, Denmark, 2009.

[47] J. Angeles. *Fundamentals of Robotic Mechanical Systems: Theory, Methods, and Algorithms.* Springer, third edition, 2007.

[48] J. M. Hollerbach. A recursive lagrangian formulation of manipulator dynamics and a comparative study of dynamics formulation complexity. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-10(11):730–736, 1980.

[49] J. Y. S. Luh, M. W. Walker, and R. P. Paul. On-line computational scheme for mechanical manipulators. *ASME Journal of Dynamic Systems, Measurement, and Control*, 120:69–76, 1980.

[50] Engineering data for harmonic drive gears. Available from: `www.harmonicdrive.de/cms/upload/pdf/en/cpu_h7.pdf`.

[51] R. L. Norten. *Machine Design: An Integrated Approach*. Prentics Hall, fourth edition, 2010.

[52] G. G. Antony. Rating and sizing of precision low backlash planetary gearboxes for automation motion control and robotics applications. Available from: `www.neugartusa.com/Service/faq/Gear_Rating.pdf`.

[53] Maxon motor products catalogue 10/11. Available from: `www.maxonmotor.ch/e-paper/blaetterkatalog/pdf/complete.pdf`.

[54] Harmonic drive technical data. Available from: `www.harmonicdrive.de/cms/upload/German/B_Produkte/B_Units/kompl_Produktkapitel_CPU_D-E.pdf`.

[55] L. Zhou, M. R. Hansen, and S. Bai. Drive train design optimization of a 5-dof light-weight robotic arm. In *Proceedings of the 1st Joint International Conference on Multibody System Dynamics*, Lappeenranta, Finland, 2010.

[56] T. Yoshikawa. Manipulability of robotic mechanisms. *The International Journal of Robotics Research*, 4(2):3–9, 1985.

[57] C. Gosselin and J. Angeles. A global performance index for the kinematic optimization of robotic manipulators. *ASME Journal of Mechanical Design*, 113:220–226, 1991.

[58] J. K. Salisbury and J. J. Craig. Articulated hands: Force control and kinematic issues. *The International Journal of Robotics Research*, 1(1):4–17, 1982.

[59] C. Gosselin and J. Angeles. The optimum kinematic design of a planar three-degree-of-freedom parallel manipulator. *ASME Journal of Mechanisms, Transmissions, and Automation in Design*, 110(1):35–41, 1988.

[60] C. Gosselin. The optimum design of robotic manipulators using dexterity indices. *Robotics and Autonomous Systems*, 9(4):213–226, 1992.

[61] S. Kucuk and Z. Bingul. Comparative study of performance indices for fundamental robot manipulators. *Robotics and Autonomous Systems*, 54:567–573, 2006.

[62] S. Bai. Optimum design of spherical parallel manipulators for a prescribed workspace. *Mechanism and Machine Theory*, 45:200–211, 2010.

[63] L. Zhou, S. Bai, and M. R. Hansen. Design optimization on the drive train of a light-weight robotic arm. *Mechatronics*, 21(3):560–569, 2011.

[64] L. Zhou, S. Bai, and M. R. Hansen. Integrated design optimization of a 5-dof assistive light-weight anthropomorphic arm. In *Proceedings of IEEE 15th International Conference on Advanced Robotics*, pages 659–664, 2011.

[65] J. A. Guin. Modification of the complex method of constrained optimization. *Computer Journal*, 10:416–417, 1968.

[66] T. Flash and N. Hogan. The coordination of arm movements: An experimentally confirmed mathematical model. *Journal of Neuroscience*, 5(7):1688–1703, July 1985.

[67] C. G. Atkeson and J. M. Hollerbach. Kinematic features of unrestrained vertical arm movements. *Journal of Neuroscience*, 5(9):2318–2330, Sep 1985.

[68] T. Kashima, Y. Isurugi, and M. Shima. An optimal control model of a neuromuscular system in human arm movements and its contrl characteristics. *Artificial Life Robotics*, 6:205–209, 2002.

[69] A. H. Fagg, A. Shah, and A. G. Barto. A computational model of muscle recruitment for wrist movements. *Journal of Neurophysiology*, 88:3348–3358, 2002.

[70] A. P. Georgopoulos, A. B. Schwartz, and R. E. Kettner. Neuronal population coding of movement direction. *Science*, 233:1416–1419, 1986.

[71] W. Abend, E. Bizzi, and P. Morasso. Human arm trajectory formation. *Brain*, 105:331–348, 1982.

[72] P. Morasso. Spatial control of arm movements. *Experimental Brain Research*, 42:223–227, 1981.

[73] D. A. Rosenbaum, L. D. Loukopoulos, R. G. J. Meulenbroek, J. Vaughan, and S. E. Engelbrecht. Planning reaches by evaluating stored postures. *Psychological Review*, 102(1):28–67, 1995.

[74] T. Kashima and Y. Isurugi. Trajectory formation based on physiological characteristics of skeletal muscles. *Biological Cybernetics*, 78:413–422, 1998.

[75] E. L. Secco, L. Valandro, R. Caimmi, G. Magenes, and B. Salvato. Optimization of two-joint arm movements: a model technique or a result of natural selection? *Biological Cybernetics*, 93:288–306, 2005.

[76] R. McN. Alexander. A minimum energy cost hypothesis for human arm trajectories. *Biological Cybernetics*, 76(2):97–105, 1997.

[77] B. R. Umberger, K. G. M. Gerritsen, and P. E. Martin. A model of human muscle energy expenditure. *Computer Methods in Biomechanics and Biomedical Engineering*, 6(2):99–111, 2003.

[78] AnyBody. AnyBody Modeling System, 2010. Available from: `www.anybodytech.com`.

[79] D. A. Winter. *Biomechanics and motor control of human movement*. John Wiley & Sons, New Jersey, 4 edition, 2009.

[80] A. V. Hill. The heat of shortening and the dynamic constants of muscle. *Proceedings of the Royal Society of London, Series B, Biological Sciences*, 126(843):136–195, 1938.

[81] K. R. S. Holzbaur, W. M. Murray, and S. L. Delp. A model of the upper extremity for simulating musculoskeletal surgery and analyzing neuromuscular control. *Annals of Biomedical Engineering*, 33(6):829–840, June 2005.

[82] D. B. Chaffin, G. B. J. Andersson, and B. J. Martin. *Occupational Biomechanics*. John Wiley & Sons, New York, 4 edition, 2006.

[83] J. Mendez and A. Keys. Density and composition of mammalian muscle. *Metabolism, Clinical and Experimental*, 9:184–188, 1960.

[84] J. L. Van Leeuwen. Optimum power output and structural design of sarcomeres. *Journal of Theoretical Biology*, 149:229–256, 1991.

[85] J. M. Winters and L. Stark. Analysis of fundamental human movement patterns through the use of in-depth antagonistic muscle models. *IEEE Transactions on Biomedical Engineering*, 32:826–839, 1985.

[86] S. Ma and G. I. Zahalak. A distribution-moment model of energetics in skeletal muscle. *Journal of Biomechanics*, 24(1):21–35, 1991.

Nomenclature

$\mathbf{q}$	position vector
$\mathbf{R}$	rotation matrix
${}^{j}A_i$	homogeneous transformation matrix
$\ddot{\theta}$	joint angular acceleration
$\dot{\theta}$	joint angular velocity
$\mathbf{J}$	Jacobian matrix
$\mathbf{v}_{ef}$	end-effector velocity
$\mathbf{v}_c$	linear velocity of the center of mass of link
$\boldsymbol{\omega}$	angular velocity of link
$\mathbf{I}$	inertia matrix
$\mathbf{M}$	mass matrix
$\boldsymbol{\xi}(\mathbf{G})$	vector of gravitational forces
$\mathbf{v}(\mathbf{V})$	vector of Coriolis and centrifugal terms
η_g	gear efficiency
$\mathbf{u}_m$	array of index numbers for motors
$\mathbf{u}_g$	array of index numbers for gearboxes
$\mathbf{x}_b(\mathbf{x}_w)$	best (worst) design point
$\mathbf{x}_c$	the centroid of all design points
$\mathbf{x}_{cand}$	candidate design point
ρ	gear ratio
τ	required joint torque
J_m	moment of inertia of a motor
J_g	moment of inertia of a gear
$m_m(m_g)$	mass of a motor (gear)
N_g^{max}	max permissible input speed of gearbox
N_m^{max}	max permissible motor speed

n_p	required motor peak speed
n_{in}	required max input peak speed of gearbox
T_g^{max}	limit for momentary peak torque of gearbox
T_m^{max}	motor stall torque
T_g	limit for rated torque of gearbox
T_m	nominal motor torque
τ_g	required gear peak torque
τ_m	required motor torque
τ_p	required motor peak torque
τ_{rmc}	RMC value of required joint torque
τ_{rms}	RMS value of required motor torque
GCI	global conditioning index
$\mathbf{C}_{min}$	minimum acceptable GCI
$\mathbf{u}_d$	array of index numbers for dimensions
$\mathbf{r}$	vector of link length ratio
S_{max}	maximum von-Mises stress
S_y	yield strength
D_{max}	maximum deflection of the end-effector
D_{lim}	acceptable deflection of the end-effector
$\dot{E}$	total energy expenditure rate of a single muscle
L_{opt}	optimal fiber length of a torque provider
M_{max}	maximum strength-moment
M_{iso}	moment generated by muscle isometric force
$M_s(M_e)$	joint moment generated at shoulder (elbow) joint
P_{mech}	mechanical power of a muscle-tendon unit
P_m	metabolic power of a muscle-tendon unit
$D_s(D_e)$	angle deviation of the shoulder (elbow) joint
E_m	metabolic energy cost